Riemannian Optimization for Convex and Non-Convex Signal Processing and Machine Learning Applications

Thesis by
Ahmed Douik

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

Caltech

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2020
Defended May 27th, 2020

ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Prof. Babak Hassibi for the continuous support of my Ph.D research, for his motivation, and immense knowledge.

Besides my advisor, I would like to thank the rest of my thesis committee, Prof. P. P. Vaidyanathan, Prof. V. Kostina, Prof. J. Tropp and, Prof. V. Chandrasekaran, for their insightful and valuable comments and encouragement.

Finally, I would like to also thank my fellow labmates and friends from Caltech for the stimulating discussions and for all the fun we have had in the last five years.

ABSTRACT

The performance of most algorithms for signal processing and machine learning applications highly depends on the underlying optimization algorithms. Multiple techniques have been proposed for solving convex and non-convex problems such as interior-point methods and semidefinite programming. However, it is well known that these algorithms are not ideally suited for large-scale optimization with a high number of variables and/or constraints. This thesis exploits a novel optimization method, known as Riemannian optimization, for efficiently solving convex and non-convex problems with signal processing and machine learning applications. Unlike most optimization techniques whose complexities increase with the number of constraints, Riemannian methods smartly exploit the structure of the search space, a.k.a., the set of feasible solutions, to reduce the embedded dimension and efficiently solve optimization problems in a reasonable time. However, such efficiency comes at the expense of universality as the geometry of each manifold needs to be investigated individually. This thesis explains the steps of designing first and second-order Riemannian optimization methods for smooth matrix manifolds through the study and design of optimization algorithms for various applications. In particular, the paper is interested in contemporary applications in signal processing and machine learning, such as community detection, graph-based clustering, phase retrieval, and indoor and outdoor location determination. Simulation results are provided to attest to the efficiency of the proposed methods against popular generic and specialized solvers for each of the above applications.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] A. Douik et al. “Precise 3-D GNSS Attitude Determination Based on Riemannian Manifold Optimization Algorithms”. In: *IEEE Transactions on Signal Processing* 68.1 (Dec. 2020), pp. 284–299. DOI: 10.1109/TSP.2019.2959226.
Ahmed Douik participated in the conception of the project, designed and implemented the optimization algorithm, helped with simulations, and wrote parts of the manuscript.

[2] A. Douik and B. Hassibi. “Manifold Optimization Over the Set of Doubly Stochastic Matrices: A Second-Order Geometry”. In: *IEEE Transactions on Signal Processing* 67.22 (Nov. 2019), pp. 5761–5774. DOI: 10.1109/TSP.2019.2946024.
Ahmed Douik participated in the conception of the project, solved the problem, designed the algorithm, prepared the simulations, and wrote the manuscript.

[3] A. Douik and B. Hassibi. “Non-Negative Matrix Factorization via Low-Rank Stochastic Manifold Optimization”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2019), Paris, France*. Vol. 1. 1. June 2019, pp. 497–501. DOI: 10.1109/ISIT.2019.8849441.
Ahmed Douik participated in the conception of the project, solved the problem, designed the algorithm, prepared the simulations, and wrote the manuscript.

[4] A. Douik, F. Salehi, and B. Hassibi. “A Novel Riemannian Optimization Approach and Algorithm for Solving the Phase Retrieval Problem”. In: *Proc. of the 53rd Asilomar Conference on Signals, Systems, and Computers (Asilomar’ 2019), Asilomar, CA, USA*. Vol. 1. 1. Nov. 2019, pp. 1962–1966. DOI: 10.1109/IEEECONF44664.2019.9049040.
Ahmed Douik participated in the conception of the project, studied the problem, designed and implemented the algorithm, helped with the simulations, and wrote parts of the manuscript.

[5] A. Douik and B. Hassibi. “A Riemannian Approach for Graph-Based Clustering by Doubly Stochastic Matrices”. In: *Proc. of the IEEE Statistical Signal Processing Workshop (SSP’ 2018), Freiburg, Germany*. Vol. 1. 1. June 2018, pp. 806–810. DOI: 10.1109/SSP.2018.8450685.
Ahmed Douik participated in the conception of the project, solved the problem, designed the algorithm, prepared the simulations, and wrote the manuscript.

[6] A. Douik and B. Hassibi. “An Improved Initialization for Low-Rank Matrix Completion Based on Rank-1 Updates”. In: *Proc. of the IEEE International Conf. on Acoustics Speech and Signal Processing (ICASSP’ 2018), Calgary,*

AL, Canada. June 2018, pp. 1–5. DOI: 10.1109/ICASSP.2018.8461826.
Ahmed Douik participated in the conception of the project, designed and implemented the initialization, prepared the simulations, and wrote the manuscript.

[7] A. Douik and B. Hassibi. “Low-Rank Riemannian Optimization on Positive Semidefinite Stochastic Matrices with Applications to Graph Clustering”. In: *Proc. of the International Conference on Machine Learning (ICML’ 2018), Stockholm, Sweden*. Vol. 80. July 2018, pp. 1299–1308. URL: proceedings.mlr.press/v80/douik18a.html.
Ahmed Douik participated in the conception of the project, solved the problem, designed the algorithm, prepared the simulations, and wrote the manuscript.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ACRONYMS

A-HALS Accelerated Hierarchical Alternating Least Squares. 87–89

A-MU Accelerated Multiplicative Updates. 87–89

A-PGM Accelerated Projected Gradient Method. 87

ALM Augmented Lagrange Multipliers. 74

ALS Alternating Least Squares. 87–89

ANLS Alternating Nonnegative Least Squares. 87–89

AoA Angle of Arrival. 110, 111, 131

ARC Adaptive Regularization with Cubics. 124

BFGS Broyden–Fletcher–Goldfarb–Shanno. 46

BHHH Berndt-Hall-Hall-Hausman. 46

CG Conjugate Gradient. 4, 74, 87

DFT Discrete Fourier Transform. 91

DoP Dilution of Precision. 148

END East-North-Down. 6

ENU East-North-Up. 6

GN Gauss-Newton. 124, 125

GNSS Global Navigation Satellite Systems. 5–8, 127–132, 134, 144, 147

GPS Global Positioning System. 147–150

GSO Gram-Schmidt Orthonormalization. 128

HALS Hierarchical Alternating Least Squares. 87–89

IoT Internet of Things. 6

IPM Interior-Point Method. 44–46, 122, 126

KKT Karush-Kuhn-Tucker. 54, 57

LRM Low-Rank Multinomial. 87–89

LS Least Squares. 132, 145–147, 149–151

LU Lower–Upper. 43

MM Majorize Minimization. 49

MSE Mean Square Error. 124–126

MU Multiplicative Updates. 87, 89

NMF Non-negative Matrix Factorization. 80, 87, 89

PGM Projected Gradient Methods. 87

PSD Positive Semidefinite. 53, 54

RAM Random Access Memory. 44, 74, 124

RMSE Root Mean Square Error. 148, 150, 151

RSS Received Signal Strength. 110

SNR Signal-to-Noise Ratio. 88, 124

SQP Sequential Quadratic Programming. 44–47, 122, 126

SVD Singular Value Decomposition. 80

TDoA Time Difference of Arrival. 112

ToA Time of Arrival. 111

ToF Time of Flight. 109–111

TR Trust Regions. 4, 20, 74, 79, 99, 106

Chapter 1

INTRODUCTION

Optimization is a fundamental and crucial tool for most signal processing and machine learning applications. Several classes of optimization have been identified in the literature, e.g., discrete [1], non-smooth [2], and derivative-free [3] optimization. For its numerous applications, this thesis only focuses on *smooth continuous* optimization, denoted simply by the generic term “optimization" in the rest of the thesis. Historically initiated with the study of least-squares and linear programming problems [4], convex optimization is an essential subclass of optimization problems in which both the objective function and the search set, i.e., constraints, are convex. As the successor and the generalization of linear programs, convex optimization received significant attention from the research community thanks to the desirable convergence property it exhibits. As a matter of fact, under mild conditions, well-defined and explicit convex problems can be solved numerically efficiently [5]. As such, it has been successfully and extensively used in numerous applications, regardless of their convexity [6]. Indeed, while the actual applications may not be convex, reformulating or approximating the problem by a convex program has been a successful approach to obtaining “good-enough" solutions. Such a strategy is known in the scientific literature as a *convex relaxation* of the problem [7].

Despite their relative success, relying on relaxations and convex solvers might lead to poor performance in some instances. Indeed, while convex relaxations can induce an unwanted degradation in the quality of the solution, it has been established that convex methods are excessively slow for high dimensions. In other words, they suffer from the curse of dimensionality. As such, it has been observed that for some contemporary signal processing and machine learning applications, non-convex solvers are significantly more efficient than their convex counterparts both in terms of quality of the solution and convergence time [8]. Such behavior is primarily due to the fact that non-convex methods successfully exploit the structure of the problem, which is destroyed in the convex reformulation/approximation. Therefore, proposing non-convex solvers has been an emerging and captivating research topic of late, e.g., see [9, 10] and references therein.

Multiple algorithms have been proposed in the literature to solve both convex and non-convex constrained optimization problems. These algorithms include the celebrated interior-point methods [11], semi-definite programming [5], Lagrangian multiplier methods [12], and simpler approaches such as alternating minimization algorithms [13, 14], including alternating orthogonal projection [15]. While these algorithms have the merit of being both generic and easily implementable, their convergence might be excessively time-consuming for high-dimensional convex problems. Furthermore, the performance of these algorithms is largely unknown for non-convex programs.

This thesis presents an alternative optimization method that circumvents both above-identified limitations, i.e., poor solution's quality and high complexity, resulting in highly efficient convex and non-convex optimization algorithms. The optimization method, known as *Riemannian optimization*, solves the constrained optimization problem as an unconstrained one over a restricted search space. This restriction of the search space allows the solution to be feasible while reducing the dimension of the problem, thus providing Riemannian optimization with exceptional abilities in finding efficient solutions in a reasonable time. However, the same mechanism introduces curvature as the search space is no longer Euclidean, which hinders the universality of the method for the geometry of each manifold needs to be investigated individually. To that end, this paper explains the steps of designing first and second-order Riemannian optimization methods for smooth matrix manifolds through the study and design of optimization algorithms for various contemporary applications in signal processing and machine learning.

1.1 Introduction to Riemannian Optimization Methods

Unconstrained and Riemannian Optimization Methods

As stated earlier, numerical optimization is the foundation of various engineering and computational sciences. Consider a smooth map f from a subset $\mathcal{D}$ of $\mathbb{R}^n$ to $\mathbb{R}$. The goal of optimization algorithms is to discover an extreme point $\mathbf{x}^* \in \mathcal{D}$ such that $\mathsf{f}(\mathbf{x}^*) \leq \mathsf{f}(\mathbf{y})$ for all feasible points $\mathbf{y} \in \mathcal{N}_{\mathbf{x}^*}$ in the neighborhood of $\mathbf{x}^*$. These traditional optimization schemes in which the embedding space is linear, such as the space of real vectors $\mathbb{R}^n$ and matrices $\mathbb{R}^{n\times m}$, are identified with the term Euclidean in contrast with the Riemannian algorithms in the rest of the paper. While unconstrained Euclidean optimization refers to the setup in which the domain of the objective function is the whole space, i.e., $\mathcal{D} = \mathbb{R}^n$, constrained Euclidean optimization denotes problems in which the search set is restricted, i.e., $\mathcal{D} \subsetneq \mathbb{R}^n$.

This manuscript is interested in the generic class of optimization problems in which the interior of the search space can be identified with a manifold that is embedded in a higher-dimensional Euclidean space, e.g., low-rank matrices embedded in the Euclidean space of matrices $\mathbb{R}^{n\times m}$. While such problems can be solved using constrained optimization methods [11], these algorithms can be excessively slow as they require solving on the high-dimensional Euclidean space. Riemannian optimization takes advantage of the fact that the manifold is of lower dimension and exploits its underlying geometric structure to reduce the computation complexity significantly.

The above property is achieved by extending unconstrained optimization schemes from Euclidean spaces to Riemannian manifolds. As such, Riemannian optimization methods fall within the scope of iterative optimization algorithms. In other words, starting from an initial position, the algorithm sequentially finds a series of neighboring points that converge to a critical point of the optimization objective function. However, unlike the interior-point method and similar algorithms that modify the objective function by including a barrier or additional (dual) variables, the philosophy of Riemannian optimization is to solve the constrained optimization problem as an unconstrained one over a restricted search space. Indeed, by reformulating the problem as a minimization over the set defined by the constraints, called the *search* or the *feasible space*, the problem can be thought of as an unconstrained optimization over a constrained set, known as the manifold.

Thanks to the aforementioned low-dimension feature, optimization over Riemannian manifolds is expected to perform more efficiently [16] than traditional optimization approaches. Therefore, a large body of literature has been dedicated to adapting traditional Euclidean optimization methods and their convergence properties to Riemannian manifolds. The rest of this section provides an overview of such rich Riemannian optimization literature along with the achieved milestones.

History, Merits, and Limitations of Riemannian Optimization

Riemannian optimization algorithms appeared in the literature for the first time in the 1970s work of Luenberger [17] by the adaptation of the standard Newton's method to Riemannian manifolds. The author demonstrated that the Riemannian version of Newton's method exhibits quadratic convergence. However, such guarantees come at a high complexity price per iteration. Indeed, finding the search direction requires computing the exact Hessian using parallel vector transport, and the step size is optimized by searching along the geodesic, i.e., straight line on a Riemannian manifold. The complexity is partially reduced for embedded submanifolds of $\mathbb{R}^n$ with the work of Gabay [18] which introduces the steepest-descent and the quasi-Newton algorithms and demonstrate their convergence for embedded submanifolds of $\mathbb{R}^n$.

The convergence results of the steepest-descent and the quasi-Newton in [18] are extended from embedded submanifolds of $\mathbb{R}^n$ to abstract Riemannian manifolds [19, 20]. Although these approaches are notably faster than the standard Newton's method, they are far from competing with established constrained optimization algorithms. The first breakthrough in the field happens with the substitution of the complex exact line-search by the effective Armijo step-size control while preserving the convergence rate. Such improvement allows the design of a unified framework for constrained and unconstrained optimization [21]. Another notable improvement for the quasi-Newton algorithms is the use of a general connection as an alternative to the canonical parallel vector transport for computing the approximate Hessian. However, while the modified algorithm conserves its global convergence property, it no longer ensures a superlinear convergence rate [22].

All above-mentioned Riemannian optimization works extend the unconstrained schemes to Riemannian manifolds by, inter alias, replacing the line-search part with the natural and intuitive search along the straight lines of the manifolds, viz., geodesics. The expression of these geodesics is acquired from the exponential map, which may be more challenging to derive than solving the original optimization problem [23]. The authors in [24] achieved the second breakthrough in Riemannian optimization by showing that Newton's method conserves its quadratic convergence rate when using a second-order approximation of the exponential map, known as a retraction.

With multiple efforts from the Riemannian optimization community, e.g., [25, 26, 27, 28], the result of [24] has been extended to first-order retractions and more-sophisticated optimization algorithms such as the CG and TR methods. These algorithms have been successfully implemented to efficiently solve various problems in communication and signal processing such as clustering or community detection [29, 30, 31, 32] and low-rank matrix completion [33, 34, 35, 36].

Thanks to its high efficiency in solving sizeable convex and non-convex programs, Riemannian optimization is slowly but surely gaining momentum in the optimization literature [16]. However, its use by the scientific community at large remains relatively limited mainly due to the non-universality of the method and the required pre-requisite knowledge of differential geometry and Riemannian manifolds. Indeed, as the technique relies on the structure of the problem and due to the lack of a systematic mechanism to design optimization algorithm over manifolds, its use might be prohibitively complicated for non-experts in the field. Nonetheless, thanks to the implementation of multiple optimization algorithms over a broad set of manifolds, e.g., the MATLAB optimization framework ManOpt [37], Riemannian optimization is becoming more and more user-friendly.

1.2 Signal Processing and Machine Learning Problems of Interest

This manuscript exploits Riemannian optimization techniques to solve multiple problems in signal processing and machine learning. These applications include convex and non-convex graph-based clustering [38], also known as the similarity clustering [39, 40, 41] by doubly stochastic matrices, Fourier phase retrieval [42, 43, 44], indoor high-accuracy spatial location and orientation estimation using ultrasound waves [45, 46, 47], and outdoor precise attitude determination in the Global Navigation Satellite Systems (GNSS) [48, 49, 50, 51, 52].

Clustering by Doubly Stochastic Matrices

Convex and non-convex optimization over the set of doubly stochastic matrices has numerous applications in signal processing and machine learning applications for its close relationship with graph theory and Markov chains. One of the most popular application of doubly stochastic matrices in machine learning is graph-based clustering [39, 53, 54, 55] wherein one wishes to separate data points into different groups, known as clusters. Similarity clustering is a subset of clustering applications in which one is given an entry-wise non-negative similarity matrix $\mathbf{A}$ between n data points with the goal of clustering these data points into r clusters. Multiple convex, e.g., [39], and non-convex, e.g., [53], approaches have been proposed to solve the similarity clustering problem. This thesis provides a unified framework to carry such optimization.

Fourier Phase Retrieval

Phase retrieval is a classical problem in signal processing [42] in which one wishes to recover a complex signal from observations of the amplitude of linear combinations of the signal. The problem is crucial in multiple imaging applications wherein the phase of the signal cannot be measured for technical or economic reasons.

Indeed, due to the inability of physical measurement devices to detect phases, e.g., a photosensitive film that measures the light intensity, only the magnitude of the signal is available. These applications include optics, crystallography, astronomical imaging, speech processing, computational biology, and blind deconvolution.

Phase retrieval engaged several researchers over the years with numerous theoretical results [43, 56] and practical algorithms [57]. This manuscript focuses on the popular case of Fourier phase retrieval [58, 59] in which a subset of the available observations are obtained through the Fourier transform of the signal. This particular structure of the phase retrieval problem allows its reformulation as a constrained optimization problem wherein the constraint set is represented by an orthonormal basis. This paper demonstrates how to effectively exploit the structure of the problem to speed up its solution.

Spatial Location Estimation Using Ultrasound Waves

With the rapidly increasing number of smartphones and the proliferation of the Internet of Things (IoT), location-based services gained an increased interest in the last decade [60]. These services range from outdoor localization, e.g., for navigation purposes, to accurate indoor pinpointing for applications such as robot steering, surveillance, video gaming, and virtual reality [61]. While outdoor localization is universally solved by the GNSS, such a system is not feasible indoors. As such, indoor localization systems have been implemented using various competing technologies, including ultrasound waves [62], radio-frequency [63], infrared radiation [64], and laser signals [65].

For most localization systems, a small perturbation in the measurement can result in a significant deviation in the expected location, especially under a bad geometry [66]. To circumvent the aforementioned limitation, this manuscript aims to design a novel and highly accurate spatial location estimation method that uses multiple transmitters and considers exploiting their geometry in the estimation process. The resulting transmitter diversity not only significantly improves the accuracy of the estimated location but also provides the 3D orientation of the device.

Attitude Determination in Global Navigation Satellite Systems

The goal of attitude determination is to estimate the orientation of a vehicle relative to the selected coordinate system, such as the East-North-Up (ENU) frame or the East-North-Down (END) frame. The 3-D attitude information can be represented using Euler angles (yaw, pitch, and roll), which can be uniquely determined by three or more non-collinear GNSS receivers. These receivers collect two types of measurements, pseudo-range and carrier phase. Carrier phase measurements are several orders of magnitude more accurate than pseudo-ranges, but are subject to ambiguities due to unknown (integer) number of unobserved wavelength cycles [67]. Therefore, phase ambiguity resolution is a critical process for high-accuracy attitude determination. After successfully resolving the ambiguity, the carrier phase can be utilized to precisely estimate direction vectors of the receiver baselines [68]. The estimation can be enhanced by leveraging geometrical information of the receiver configuration, as well as the unitary nature of the direction vectors of interest. This manuscript uses Riemannian optimization techniques to obtain the baseline's direction vectors, which can be directly converted into Euler angles to fully characterize the attitude of the platform.

1.3 Contributions, Notations, and Organization

Scope and Contributions

The main contribution of this paper is to exploit Riemannian optimization methods to design efficient optimization algorithms to solve the aforementioned convex and non-convex problems in signal processing and machine learning. However, such endeavor requires some level of knowledge of differential and Riemannian geometries. To that end, the manuscript provides an introduction on designing efficient first and second-order Riemannian optimization methods for smooth matrix manifolds. As only smooth embedded and quotient matrix manifolds are considered, the definitions and theorems herein may not apply to abstract manifolds. In addition, the author opted for a coordinate-free analysis omitting charts and differentiable structures of manifolds. For an introduction to differential geometry, abstract manifolds, and Riemannian manifolds, we refer the readers to the following references [69, 70, 71], respectively.

After introducing the necessary machinery to design Riemannian optimization algorithm, this thesis investigates the aforementioned convex and non-convex clustering, phase retrieval, and localization applications. In particular, the doubly stochastic and symmetric stochastic multinomial manifolds are introduced and their geometries investigated so as to design efficient convex clustering algorithms. Afterwards, the low-rank property of the solution to the clustering problem is exploited to reformulate the community detection task as a non-convex program. The non-convex reformulation is solved by providing theoretical guarantees on the quality of the solution and investigating the geometry of the low-rank symmetric stochastic multinomial.

For the phase retrieval problem, the feasible set is represented by generic quadratic equations. While there is no guarantees that such set admits a manifold structure in general, the manuscript shows that the set of feasible solutions represents a Riemannian manifold under Fourier observations. The first and second-order geometries of the newly introduced manifold, known as the fixed norms manifold, are investigated and efficient phase retrieval algorithms are designed. Furthermore, by exploiting the fact that the optimization problem for phase retrieval presents non-isolated solutions, the manifold is demonstrated to admit a quotient structure, allowing the design of even faster algorithms.

Afterwards, the thesis investigates and designs indoor and outdoor precise localization systems. For instance, the problem of accurate indoor spatial location and orientation estimation using ultrasound waves is considered. To improve the accuracy of the localization system, the transmitters' geometry is integrated into the location estimation process by formulating the problem as a non-convex optimization. Afterward, the set of feasible solutions is shown to admit a Riemannian manifold structure, which allows solving the underlying optimization problem rigorously.

Finally, outdoor localization is examined through the problem of altitude determination in GNSS. The antenna geometry and baseline lengths are exploited to formulate the 3-D GNSS attitude determination problem as an optimization over a non-convex set, shown to be a manifold. The study of the geometry of the manifold allows the design of efficient first and second-order Riemannian algorithms to solve the 3-D GNSS attitude determination problem.

Organization

The rest of this manuscript is organized as follows. Chap. 2 introduces Riemannian manifolds and optimization techniques. These techniques are utilized in Chap. 3 to design efficient optimization algorithms for convex graph-based clustering. The results are extended to low-rank matrices in Chap. 4 so as to tackle non-convex community detection. The Fourier phase retrieval problem is investigated in Chap. 5. Finally, before concluding in Chap. 7, accurate indoor and outdoor localization algorithms are designed in Chap. 6.

Notations

Throughout the paper, lower-case letters x denotes scalar variables, while boldface lower-case letters $\mathbf{x}$ and boldface upper-case $\mathbf{X}$ denote vectors and matrices, respectively. Tangent vectors and matrices are an exception to the previous rule and are denoted by Greek letters with an index representing the foot of the tangent space, e.g., $\xi_{\mathbf{x}}$ represents a tangent vector at $\mathbf{x}$. These Riemannian geometry related notations would become more apparent in Chap. 2.

The i-th entry of vector $\mathbf{x}$ is denoted by $\mathbf{x}_i$ and the element in the i-th row and j-th column of matrix $\mathbf{X}$ is denoted by $\mathbf{X}_{ij}$. The notations $\mathbf{0}$, $\mathbf{1}$, and $\mathbf{I}$ denote the null vector, the all ones vector, and the identity matrix, respectively. If the dimension of the vector is not clear from the context, it is added as a subscript. For example, $\mathbf{I}_n$ denote the identity matrix of size $n \times n$.

Given a vector $\mathbf{x}$, the symbols $|\mathbf{x}|$, $\mathbf{x}^{\mathrm{T}}$, and $\mathbf{x}^{\star}$ denote the absolute value, the transpose and the conjugate transpose operators of vector $\mathbf{x}$, respectively. The absolute value of a vector is defined as the absolute value of each of its entries. For a matrix $\mathbf{X}$, the notations $||\mathbf{X}||$, $\mathbf{X}^{\mathrm{T}}$, and $\mathbf{X}^{\mathrm{H}}$ denotes the Frobenius norm, the transpose and the Hermitian operators of matrix $\mathbf{X}$, respectively.

While sets are denoted by a calligraphic font, e.g., $\mathcal{X}$, the sets of real and complex numbers are denoted by $\mathbb{R}$ and $\mathbb{C}$, respectively. The set of real and complex vectors and matrices follow the usual notation of placing the dimension as exponents. For instance, $\mathbb{R}^{n\times m}$ refers to the set of real matrices of size $n \times m$. Given a symmetric (or Hermitian) matrix $\mathbf{X} = \mathbf{X}^{\mathrm{T}}$ (or $\mathbf{X} = \mathbf{X}^{\mathrm{H}}$), the notation $\mathbf{X} \succ \mathbf{0}$ means that matrix is positive-definite, i.e., all its eigenvalues are strictly positive. Likewise, the symbol $\mathbf{X} > \mathbf{0}$ refers to a real matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$ with strictly positive entries, i.e., $\mathbf{X}_{ij} > 0$ for all $1 \leq i \leq n$ and $1 \leq j \leq m$.

Let the notation $\mathsf{Tr}(.)$ refer to the trace operator and $\langle \mathbf{X}, \mathbf{Y} \rangle = \mathsf{Tr}(\mathbf{Y}^{\mathrm{T}}\mathbf{X})$ to the Frobenius inner product of matrices $\mathbf{X}$ and $\mathbf{Y}$ on the space $\mathbb{R}^{n\times n}$. Given two matrices, the element-wise product, a.k.a., the Hadamard product, is denoted by the symbol $\odot$, i.e., $(\mathbf{X} \odot \mathbf{Y})_{ij} = \mathbf{X}_{ij}\mathbf{Y}_{ij}$. Similarly, the Hadamard division is denoted by the symbol $\oslash$. Let $\mathcal{S}^n$ and $\mathcal{S}^n_{\mathsf{skew}}$ be the set of symmetric and skew-symmetric matrices, respectively. A full-rank $n \times p$ matrix $\mathbf{Y}$ is an element of the set $\mathbb{R}^{n\times m}_{\star}$. The set $\mathcal{O}^p = \{\mathbf{O} \in \mathbb{R}^{p\times p}_{\star} | \mathbf{O}\mathbf{O}^{\mathrm{T}} = \mathbf{I}\}$ represents orthogonal matrices.

Functions are denoted by lower- and upper-case letters with sans serif font. Additionally, Greek letters are used to denote functions representing curves on manifolds.

To explicitly mention the pre-image and image sets by a function F, the standard notation $\mathsf{F} : \mathcal{X} \longrightarrow \mathcal{Y}$ is used. For a single variable function, e.g., $\gamma(t)$, the shorthand notation $\dot{\gamma}(t)$ is used to denote the first-order derivative $\dot{\gamma}(t) = \frac{\mathrm{d}\gamma}{\mathrm{d}t}$.

The set of continuously differentiable is denoted by C^1. Similarly, the symbol C^2 is used to refer to the class of all first and second continuously differentiable functions. In the rest of the manuscript, a smooth function refers to a function of class at least C^1 for first-order algorithms and at least C^2 for second-order methods.

Chapter 2

OPTIMIZATION ON RIEMANNIAN MANIFOLDS

This chapter introduces Riemannian optimization methods on embedded and quotient Riemannian matrix manifolds. In particular, Section 2.1 presents the manifold definitions and notations used throughout the thesis. Afterward, the geometries of embedded and quotient matrix submanifolds are investigates in Sections 2.2 and 2.3, respectively. Finally, Section 2.4 exploits such geometries to design various first and second-order Riemannian optimization algorithms.

2.1 Manifold Definitions and Notation

The philosophy of Riemannian optimization techniques is to extend unconstrained optimization methods from Euclidean spaces to manifolds. As such, this part first recalls the principals of Euclidean optimization. Afterward, Riemannian matrix manifolds and submanifolds are introduced. Finally, an overview of Riemannian optimization methods is presented. For clarity purposes, the optimization over linear spaces, i.e., Euclidean spaces, is refereed as Euclidean optimization with contrast with Riemannian optimization.

Euclidean Spaces and Optimization

The general idea behind unconstrained Euclidean numerical optimization methods is to start with an initial point $\mathbf{X}^0$ and to iteratively update it according to certain predefined rules in order to obtain a sequence $\{\mathbf{X}^t\}_{t=0}^{\infty}$ which converges to a local minimum of the objective function. A typical update strategy is $\mathbf{X}^{t+1} = \mathbf{X}^t + \alpha^t p^t$ where α^t is the step size and p^t the search direction. Let $\mathsf{Grad}\ \mathsf{f}(\mathbf{X})$ be the Euclidean gradient of the objective function defined as the unique vector satisfying:

$$\langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \xi \rangle = \mathsf{D}\ \mathsf{f}(\mathbf{X})[\xi],\ \forall\, \xi \in \mathcal{E},$$

where $\langle ., . \rangle$ is the inner product on the vector space $\mathcal{E}$ and $\mathsf{D}\ \mathsf{f}(\mathbf{X})[\xi]$ is the directional derivative of f given by:

$$\mathsf{D}\ \mathsf{f}(\mathbf{X})[\xi] = \lim_{t \to 0} \frac{\mathsf{f}(\mathbf{X} + t\xi) - \mathsf{f}(\mathbf{X})}{t}.$$

In order to obtain a descent direction, i.e., $\mathsf{f}(\mathbf{X}^{t+1}) < \mathsf{f}(\mathbf{X}^t)$ for a small-enough step size α^t, the search direction p^t is chosen in the half space spanned by $-\mathsf{Grad}\ \mathsf{f}(\mathbf{X})$.

In other words, the following inequality holds:

$$\langle \mathsf{Grad\ f}(\mathbf{X}^t), p^t \rangle < 0. \tag{2.1}$$

In particular, the choices of the search direction satisfying

$$p^t = -\frac{\mathsf{Grad\ f}(\mathbf{X}^t)}{||\mathsf{Grad\ f}(\mathbf{X}^t)||} \tag{2.2}$$

$$\mathsf{Hess\ f}(\mathbf{X}^t)[p^t] = \mathsf{Grad\ f}(\mathbf{X}), \tag{2.3}$$

yields the celebrated steepest-descent (2.2) and the Newton's method (2.3), wherein $\mathsf{Hess\ f}(\mathbf{X})[\xi]$ is the Euclidean Hessian of f at $\mathbf{X}$ defined as an operator from $\mathcal{E}$ to $\mathcal{E}$ satisfying:

1. $\langle \mathsf{Hess\ f}(\mathbf{X})[\xi], \xi \rangle = \mathsf{D}^2\ \mathsf{f}(\mathbf{X})[\xi, \xi] = \mathsf{D}\ (\mathsf{D\ f}(\mathbf{X})[\xi])[\xi]$,
2. $\langle \mathsf{Hess\ f}(\mathbf{X})[\xi], \eta \rangle = \langle \xi, \mathsf{Hess\ f}(\mathbf{X})[\eta] \rangle,\ \forall\, \xi, \eta \in \mathcal{E}$.

Riemannian Matrix Manifolds

A matrix manifold $\mathcal{M}$ is a smooth subset of a vector space $\mathcal{E}$ included in the set of matrices $\mathbb{R}^{n \times m}$. The set $\mathcal{E}$ is called the ambient or the embedding space. By smooth subset, we mean that $\mathcal{M}$ can be mapped by a bijective function, i.e., a chart, to an open subset of $\mathbb{R}^d$ where d is called the dimension of the manifold. The dimension d roughly represents the *degrees of freedom* of the manifold. In particular, a linear space $\mathcal{E}$ is a manifold.

Unconstrained optimization exploits both the linear structure of the embedding space and the derivatives of the function to optimize. Therefore, to generalize unconstrained algorithms, such as gradient descent and Newton's method, one needs a notion of linear approximation to a curved surface, i.e., manifold, and the concepts of gradient and Hessian on such a surface. The linearization of a smooth manifold $\mathcal{M}$ can be accomplished locally around any point $\mathbf{X} \in \mathcal{M}$ using the notion of a tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$. By endowing each tangent space with a smoothly varying inner product, known as the Riemannian metric, the manifold turns into a Riemannian manifold. Such Riemannian structure allows the definition of derivative operators similar to the gradient and Hessian and called the Riemannian gradient and Riemannian Hessian, respectively. It is implicitly understood that for an Euclidean space, the Euclidean and Riemannian gradients and Hessians coincide.

Define a real and smooth function $\mathsf{f} : \mathcal{M} \to \mathbb{R}$. The function that associates to each $\xi_{\mathbf{X}}$ the directional derivative $\mathsf{D\ f}(\mathbf{X})[\xi_{\mathbf{X}}]$ is called the indefinite directional

Table 2.1: Riemannian embedded and quotient manifolds notations.

Variable	Definition
$\mathcal{M}, \overline{\mathcal{M}}$	Embedded manifold and its quotient
$\mathbf{X}$ and $\overline{\mathbf{X}} = [\mathbf{X}]$	A point on $\mathcal{M}$ and its class on $\overline{\mathcal{M}}$
$\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$	A point on the tangent of $\mathcal{M}$
$\mathcal{H}_{\mathbf{X}}\mathcal{M}, \mathcal{V}_{\mathbf{X}}\mathcal{M}$	Horizontal and vertical spaces at $\mathbf{X}$
$\Pi_{\mathbf{X}}$	Orthogonal projection onto $\mathcal{T}_{\mathbf{X}}\mathcal{M}$
$\Pi_{\mathbf{X}}^{\mathcal{H}}$	Orthogonal projection onto $\mathcal{H}_{\mathbf{X}}\mathcal{M}$
$\Pi_{\mathbf{X}}^{\mathcal{V}}$	Orthogonal projection onto $\mathcal{V}_{\mathbf{X}}\mathcal{M}$
$\xi_{\overline{\mathbf{X}}} \in \mathcal{T}_{\overline{\mathbf{X}}}\overline{\mathcal{M}}$	A point on the tangent of $\overline{\mathcal{M}}$
$\bar{\xi}_{\mathbf{X}} \in \mathcal{H}_{\mathbf{X}}\mathcal{M}$	Horizontal lift of $\xi_{\overline{\mathbf{X}}}$ at $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$
$\mathsf{Grad}\ \mathsf{f}(\mathbf{X})$	Euclidean Gradient at $\mathbf{X}$
$\mathsf{Hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$	Euclidean Hessian at $\mathbf{X}$ and $\xi_{\mathbf{X}}$
$\mathsf{grad}\ \mathsf{f}(\mathbf{X})$	Riemannian Gradient at $\mathbf{X}$
$\mathsf{hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$	Riemannian Hessian at $\mathbf{X}$ and $\xi_{\mathbf{X}}$
$\mathsf{grad}\ \bar{\mathsf{f}}(\mathbf{X})$	Lift of $\mathsf{grad}\ \bar{\mathsf{f}}(\overline{\mathbf{X}})$ at $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$
$\mathsf{hess}\ \bar{\mathsf{f}}(\mathbf{X})[\xi_{\mathbf{X}}]$	Lift of $\mathsf{hess}\ \bar{\mathsf{f}}(\overline{\mathbf{X}})[\xi_{\mathbf{X}}]$ at $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$
$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}})$	Retraction of $\xi_{\mathbf{X}}$ at $\mathbf{X} \in \mathcal{M}$
$\overline{\mathsf{R}}_{\overline{\mathbf{X}}}(\xi_{\overline{\mathbf{X}}})$	Retraction of lift $\bar{\xi}_{\mathbf{X}}$ at $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$

derivative of f at $\mathbf{X}$. The Euclidean and Riemannian gradients of f at $\mathbf{X} \in \mathcal{M}$ are denoted by $\mathsf{Grad}\ \mathsf{f}(\mathbf{X})$ and $\mathsf{grad}\ \mathsf{f}(\mathbf{X})$, respectively. Similarly, the Euclidean and Riemannian Hessian of f at the point $\mathbf{X} \in \mathcal{M}$ in the direction $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ are denoted by $\mathsf{Hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$ and $\mathsf{hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$, respectively.

In the rest of the manuscript, variables relative to quotient manifolds, e.g., equivalence classes, are denoted by overline characters and their representatives in the embedded manifold are represented without the overline. For convenience, Table 2.1 summarizes all Riemannian notations used in this paper. These notations are introduced in Sections 2.2 and 2.3 for the embedded and quotient submanifolds, respectively.

Overview of Riemannian Optimization Techniques

In the XIX century, Riemann investigated curvature in high-dimensional spaces which led to the developed of an abstract geometry, known today as differential and Riemannian geometries. Nowadays, these abstract geometric concepts found real applications in the realm of numerical optimization. This is accomplished by exploiting the geometry of the manifolds to extend unconstrained methods from

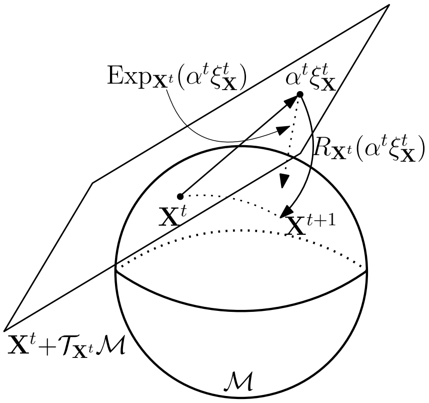

$\mathrm{Exp}_{\mathbf{X}^t}(\alpha^t\xi^t_{\mathbf{X}})$
$\alpha^t\xi^t_{\mathbf{X}}$
$R_{\mathbf{X}^t}(\alpha^t\xi^t_{\mathbf{X}})$
$\mathbf{X}^t$
$\mathbf{X}^{t+1}$
$\mathbf{X}^t+\mathcal{T}_{\mathbf{X}^t}\mathcal{M}$
$\mathcal{M}$

Figure 2.1: The update step for the two-dimensional sphere embedded in $\mathbb{R}^3$.

Euclidean to Riemannian spaces. This part explains the general concept of designing optimization algorithms using Riemannian geometry.

The fundamental idea of optimization algorithms on manifolds is to locally approximate the manifold by a linear space known as the *tangent space*. Afterwards, unconstrained optimization is performed on that tangent space. In particular, a descent direction is computed by deriving the *Riemannian gradient*. Finally, the point on the tangent space is “projected" to the manifold using a *retraction*. The steps of the algorithm are available in Algorithm 2.1 and an illustration of one iteration of the algorithm is given in Figure 2.1.

Algorithm 2.1 Template of optimization algorithms on Riemannian manifolds.

```
Require: Manifold $\mathcal{M}$, function $\mathsf{f}$, and retraction $\mathsf{R}$.
Initialize $\mathbf{X} \in \mathcal{M}$.
while $||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}} \geq \epsilon$ do
    Choose search direction $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$.
    Compute step size $\alpha$.
    Retract $\mathbf{X} = \mathsf{R}_{\mathbf{X}}(\alpha\xi_{\mathbf{X}})$.
end while
Output $\mathbf{X}$.
```

Sections 2.2 and 2.3 define the above relevant concepts from differential and Riemannian geometry for embedded and quotient manifolds, respectively. These geometric operators are exploited afterwards in Section 2.4 to design various first and second-order optimization algorithms.

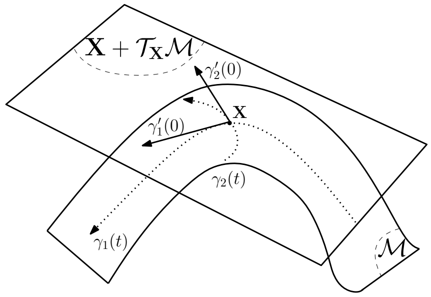


Figure 2.2: Tangent space of a 2-dimensional manifold embedded in $\mathbb{R}^3$.

2.2 Embedded Riemannian Manifolds

First-Order Embedded Manifolds Geometry

Along the lines of approximating a function locally by its derivatives, a manifold $\mathcal{M}$ of dimension d can be approximated locally at a point $\mathbf{X}$ by a d-dimensional vector space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ generated by taking derivatives of all smooth curves going through $\mathbf{X}$ at the origin. Formally, let $\gamma(t) : \mathcal{I} \subset \mathbb{R} \longrightarrow \mathcal{M}$ be a curve on $\mathcal{M}$ with $\gamma(0) = \mathbf{X}$ and denote by $\gamma'(0)$ its derivative at 0. The space generated by all such $\gamma'(0)$ represents a vector space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ called the tangent space of $\mathcal{M}$ at $\mathbf{X}$. Figure 2.2 shows an example of a two-dimensional tangent space generated by a couple of curves. The tangent space plays a primordial role in Riemannian optimization algorithms in the same way that derivatives of functions play an important role in Euclidean optimization. The union of all tangent spaces $\mathcal{T}\mathcal{M}$ is referred to as the tangent bundle of $\mathcal{M}$, i.e.,:

$$\mathcal{T}\mathcal{M} = \bigcup_{\mathbf{X}\in\mathcal{M}} \mathcal{T}_{\mathbf{X}}\mathcal{M}.$$

To optimize functions on manifolds, besides the notion of a tangent space described above, one needs the notion of directions and lengths which can be achieved by endowing each tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ by an inner product $\langle \xi_{\mathbf{X}}, \eta_{\mathbf{X}} \rangle_{\mathbf{X}}$, $\forall\, \xi_{\mathbf{X}}, \eta_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$. Such metric, known as the Riemannian metric, turns the manifold into a Riemannian manifold. The norm on the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ is denoted by $||.||_{\mathbf{X}}$ and defined by:

$$||\xi_{\mathbf{X}}||_{\mathbf{X}} = \sqrt{\langle \xi_{\mathbf{X}}, \xi_{\mathbf{X}} \rangle_{\mathbf{X}}}, \ \forall\, \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}.$$

Since both the ambient space and the tangent space are vector spaces, one can define the orthogonal projection $\Pi_{\mathbf{X}} : \mathcal{E} \longrightarrow \mathcal{T}_{\mathbf{X}}\mathcal{M}$ verifying $\Pi_{\mathbf{X}} \circ \Pi_{\mathbf{X}} = \Pi_{\mathbf{X}}$. The projection is said to be orthogonal with respect to the restriction of the Riemannian metric to the tangent space, i.e., $\Pi_{\mathbf{X}}$ is orthogonal in the $\langle ., . \rangle_{\mathbf{X}}$ sense which means that $\langle \Pi_{\mathbf{X}}(\mathbf{Y}), \mathbf{Y} - \Pi_{\mathbf{X}}(\mathbf{Y}) \rangle_{\mathbf{X}} = 0, \ \forall\, \mathbf{Y} \in \mathcal{E}$.

The Riemannian gradient is defined in a similar manner as the Euclidean one with the exception that it uses the Riemannian geometry, i.e.,:

Definition 2.1 *The Riemannian gradient of* f *of a manifold* $\mathcal{M}$ *at* $\mathbf{X}$*, denoted by* $\mathsf{grad}\ \mathsf{f}(\mathbf{X})$*, is defined as the unique tangent vector in* $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ *that satisfies:*

$$\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}), \xi_{\mathbf{X}} \rangle_{\mathbf{X}} = \mathsf{D}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}],\ \forall\, \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}.$$

While the update step $\mathbf{X}^{t+1} = \mathbf{X}^t + \alpha^t p^t$ is trivial in Euclidean optimization thanks to its vector space structure, it might result in a point $\mathbf{X}^{t+1}$ outside the manifold. Moving in a given tangent direction while remaining on the manifold is realized by the retraction operator. The ideal retraction is the exponential map[1] $\mathrm{Exp}_{\mathbf{X}}$ as it maps a tangent vector $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ to a point on the manifold along the geodesic curve (straight line on the manifold) that goes through $\mathbf{X}$ in the direction of $\xi_{\mathbf{X}}$. However, computing geodesic curves is challenging and may be more difficult that solving the original optimization problem. Luckily, one can use a first-order approximation of the exponential map, called a retraction herein, without compromising the convergence properties of the algorithms.

Definition 2.2 *A retraction on a manifold* $\mathcal{M}$ *is a smooth map* R *from the tangent bundle* $\mathcal{T}\mathcal{M}$ *onto* $\mathcal{M}$*. For all* $\mathbf{X} \in \mathcal{M}$*, the restriction of* R *to* $\mathcal{T}_{\mathbf{X}}\mathcal{M}$*, denoted by* $\mathsf{R}_{\mathbf{X}}$*, satisfies the following properties:*

- *Centering:* $\mathsf{R}_{\mathbf{X}}(0) = \mathbf{X}$.
- *Local rigidity: The curve* $\gamma_{\xi_{\mathbf{X}}}(\tau) = \mathsf{R}_{\mathbf{X}}(\tau\xi_{\mathbf{X}})$ *satisfy* $\dot{\gamma}_{\xi_{\mathbf{X}}}(0) = \xi_{\mathbf{X}},\ \forall\, \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$.

Embedded Submanifolds: Second-Order Geometry

Generalizing Newton's method to the Riemannian setting requires computing the Riemannian Hessian which can be accomplished by taking a directional derivative of a vector field. As vectors belong to different tangent spaces, one needs the notion of a connection ∇, also called the covariant derivative, which generalizes the notion of directional derivative to vector fields. The definition of a connection is given below:

[1] The exponential map and geodesics are introduced later as part of the second-order geometry of the manifold.

Definition 2.3 *An affine connection ∇ is a map that associate to (η, ξ) the tangent vector $\nabla_\eta \xi$ satisfying for all $\eta, \xi, \chi \in \mathcal{T}\mathcal{M}$, for all smooth $\mathsf{f}, \mathsf{g} : \mathcal{M} \to \mathbb{R}$, and for all reals $a, b \in \mathbb{R}$:*

- $\nabla_{\mathsf{f}(\eta)+\mathsf{g}(\chi)}\xi = \mathsf{f}(\nabla_\eta \xi) + \mathsf{g}(\nabla_\chi \xi)$
- $\nabla_\eta(a\xi + b\varphi) = a\nabla_\eta \xi + b\nabla_\eta \varphi$
- $\nabla_\eta(\mathsf{f}(\xi)) = \xi(f)\eta + \mathsf{f}(\nabla_\eta \xi),$

wherein the vector field ξ acts on the function f by derivation, i.e., $\xi(f) = \mathsf{D}\ (\mathsf{f})[\xi]$ also noted as ξf in the literature.

On a Riemannian manifold, the Levi-Civita connection is the canonical choice of affine connections as it preserves the Riemannian metric. Indeed, the Levi-Civita connection is the unique affine connection on $\mathcal{M}$ with the Riemannian metric $\langle ., . \rangle$ that satisfies for all $\eta, \xi, \chi \in \mathcal{T}\mathcal{M}$:

1. $\nabla_\eta \xi - \nabla_\xi \eta = [\eta, \xi]$
2. $\chi\langle \eta, \xi \rangle = \langle \nabla_\chi \eta, \xi \rangle + \langle \eta, \nabla_\chi \xi \rangle,$

where $[\xi, \eta]$ is the Lie bracket, i.e., a function from the set of smooth function to itself defined by $[\xi, \eta]\mathsf{g} = \xi(\eta(\mathsf{g})) - \eta(\xi(\mathsf{g}))$.

Definition 2.4 *The Riemannian Hessian of f at $\mathbf{X}$, denoted by $\mathsf{hess}\,\mathsf{f}(\mathbf{X})$, of a manifold $\mathcal{M}$ is a map from $\mathcal{T}_\mathbf{X}\mathcal{M}$ into itself defined by:*

$$\mathsf{hess}\,\mathsf{f}(\mathbf{X})[\xi_\mathbf{X}] = \nabla_{\xi_\mathbf{X}} \mathsf{grad}\,\mathsf{f}(\mathbf{X}),\ \forall\, \xi_\mathbf{X} \in \mathcal{T}_\mathbf{X}\mathcal{M},$$

where $\mathsf{grad}\,\mathsf{f}(\mathbf{X})$ is the Riemannian gradient and ∇ is the Riemannian connection on $\mathcal{M}$.

Given a Riemannian connection ∇ on $\mathcal{M}$ and an interval $\mathcal{I} \subseteq \mathbb{R}$ containing 0, a geodesic curve $\gamma : \mathcal{I} \to \mathcal{M}$ going through $\mathbf{X} \in \mathcal{M}$ in the direction $\xi_\mathbf{X} \in \mathcal{T}_\mathbf{X}\mathcal{M}$, i.e., $\gamma(0) = \mathbf{X}$ and $\dot{\gamma}(0) = \xi_\mathbf{X}$, is denoted by $\gamma_{\mathbf{X},\xi_\mathbf{X}}(t)$. The geodesic $\gamma_{\mathbf{X},\xi_\mathbf{X}}(t)$ defines the Exponential map $\mathrm{Exp}_\mathbf{X} : \mathcal{T}_\mathbf{X}\mathcal{M} \to \mathcal{M}$ by $\mathrm{Exp}_\mathbf{X}(\xi_\mathbf{X}) = \gamma_{\mathbf{X},\xi_\mathbf{X}}(1)$. In addition, if there exists a unique geodesic $\gamma_{\mathbf{X},\xi_\mathbf{X}}(t)$ between the points $\mathbf{X}$ and $\gamma_{\mathbf{X},\xi_\mathbf{X}}(1) = \mathbf{Y} \in \mathcal{M}$, the inverse of the Exponential map, also known as the Logarithmic map, is denoted by

$\mathrm{Log}_{\mathbf{X}}(\mathbf{Y}) = \mathrm{Exp}_{\mathbf{X}}^{-1}(\mathbf{Y}) = \xi_{\mathbf{X}}$. Under the previous assumption, the geodesic distance on $\mathcal{M}$, i.e., shortest distance, between points $\mathbf{X}$ and $\mathbf{Y}$ is defined and denoted by $d(\mathbf{X}, \mathbf{Y}) = ||\mathrm{Log}_{\mathbf{X}}(\mathbf{Y})||_{\mathbf{X}} = ||\mathrm{Log}_{\mathbf{Y}}(\mathbf{X})||_{\mathbf{Y}}$. Given two points $\mathbf{X}$ and $\mathbf{Y}$ in $\mathcal{M}$ and the geodesic $\gamma_{\mathbf{X},\xi_{\mathbf{X}}}(1) = \mathbf{Y}$, the parallel translation $\Gamma_{\mathbf{X}}^{\mathbf{Y}} : \mathcal{T}_{\mathbf{X}}\mathcal{M} \rightarrow \mathcal{T}_{\mathbf{Y}}\mathcal{M}$ of the tangent vector $\eta \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ is denoted by $\Gamma_{\mathbf{X}}^{\mathbf{Y}}\eta \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}$.

2.3 Quotient Riemannian Manifolds

Equivalence Relationship and Quotient Structure

Let $\sim$ be an equivalence relationship and define the set $\overline{\mathcal{M}} = \mathcal{M}/\sim$ as the quotient of the manifold $\mathcal{M}$ by $\sim$. In order to show that the set $\overline{\mathcal{M}}_p^n = \mathcal{M}_p^n/\sim$ admits a manifold structure, it is sufficient to show that $\sim$ is regular [23], meaning that it satisfies the three following properties

1. graph$(\sim)$ is an embedded submanifold of the product $\mathcal{M}_p^n \times \mathcal{M}_p^n$.
2. The projection π_1 : graph $(\sim) \rightarrow \mathcal{M}_p^n$ given by $\pi_1(\mathbf{X}_1, \mathbf{X}_2) = \mathbf{X}_1$ is a submersion.
3. graph $(\sim)$ is closed,

wherein graph $(\sim) = \{(\mathbf{X}_1, \mathbf{X}_2) \in \mathcal{M}_p^n \times \mathcal{M}_p^n \mid \mathbf{X}_1 \sim \mathbf{X}_2\}$.

Combining the three properties above allows concluding that $\overline{\mathcal{M}}_p^n$ admits a unique manifold structure known as the quotient manifold of $\mathcal{M}_p^n$ by $\sim$. However, it does not allow to conclude that $\overline{\mathcal{M}}_p^n$ inherits the Riemannian structure of $\mathcal{M}_p^n$ as it requires the Riemannian metric to be compatible with $\sim$, as described in the next part.

Under the above assumptions, the quotient manifold $\overline{\mathcal{M}}$ admits a quotient structure that groups all elements of $\mathcal{M}$ in the same equivalence class as a single point. Let π be the natural projection that associates to each $\mathbf{X} \in \mathcal{M}$ its equivalence class $\pi(\mathbf{X}) = [\mathbf{X}] = \overline{\mathbf{X}} \in \mathcal{M}$. These three notations for equivalence classes are used interchangeably in this paper.

Quotient Submanifolds: First-Order Geometry

Let $\langle ., . \rangle_{\mathbf{X}}$ be the Riemannian metric on the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ of the embedding space $\mathcal{M}$. The quotient $\overline{\mathcal{M}} = \mathcal{M}/\sim$ admits a Riemmanian structure for the induced Riemannian metric if and only if the metric is compatible with the equivalence relationship $\sim$, i.e., it does not depend on the chosen representative of the equivalence class.

To express the compatibility of the metric, we first introduce the horizontal lift. For a point $\overline{\mathbf{X}} \in \overline{\mathcal{M}}$, let $\xi_{\overline{\mathbf{X}}} \in \mathcal{T}_{\overline{\mathbf{X}}}\overline{\mathcal{M}}$ be a tangent vector. In a similar manner that $\overline{\mathbf{X}}$ can be represented by multiple $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$, the tangent vector $\xi_{\overline{\mathbf{X}}}$ can be represented by multiple predecessors for each $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$. Indeed, fix $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$, then any tangent vector $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ satisfying $\mathsf{D}\left(\pi(\mathbf{X})\right)[\xi_{\mathbf{X}}] = \xi_{\overline{\mathbf{X}}}$ can be considered as a valid representation of the tangent vector $\xi_{\overline{\mathbf{X}}}$. To circumvent the aforementioned problem and obtain a unique representation of $\xi_{\overline{\mathbf{X}}}$ for each predecessor $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$, we use the fact that $\pi^{-1}(\overline{\mathbf{X}})$ represents a manifold. Therefore, one can obtain a unique representation by orthogonally decomposing the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ into a vertical space $\mathcal{V}_{\mathbf{X}}\mathcal{M}$ and a horizontal space $\mathcal{H}_{\mathbf{X}}\mathcal{M}$ such that

$$\begin{aligned} \mathcal{V}_{\mathbf{X}}\mathcal{M} &= \mathcal{T}_{\mathbf{X}}\pi^{-1}(\overline{\mathbf{X}}) \\ \mathcal{T}_{\mathbf{X}}\mathcal{M} &= \mathcal{V}_{\mathbf{X}}\mathcal{M} \oplus \mathcal{H}_{\mathbf{X}}\mathcal{M}. \end{aligned}$$

The ambient vector space can be composed into a tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ and its orthogonal complement $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}$. In particular, for each $\mathbf{X} \in \mathcal{M}$, the embedding space $\mathbb{R}^{n\times m}$ can be uniquely decomposed into a direct sum of the above defined linear space, i.e.,

$$\mathbb{R}^{n\times m} = \mathcal{H}_{\mathbf{X}}\mathcal{M} \oplus \mathcal{V}_{\mathbf{X}}\mathcal{M} \oplus \mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}.$$

The representation of $\xi_{\overline{\mathbf{X}}} \in \mathcal{T}_{\overline{\mathbf{X}}}\overline{\mathcal{M}}$ at $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$, denoted by $\overline{\xi}_{\mathbf{X}}$ and referred to as the horizontal lift of a tangent vector $\xi_{\overline{\mathbf{X}}}$ at $\mathbf{X}$, is the unique element in the horizontal space $\mathcal{H}_{\mathbf{X}}\mathcal{M}$ satisfying $\mathsf{D}\left(\pi(\mathbf{X})\right)[\overline{\xi}_{\mathbf{X}}] = \xi_{\overline{\mathbf{X}}}$. Such representation as horizontal lift allows to get a unique parameterization of tangent vectors in a quotient manifold. The manifold $\overline{\mathcal{M}}$ represents a Riemannian manifold for the Riemmanian metric $\langle ., . \rangle_{\overline{\mathbf{X}}}$ on $\mathcal{T}_{\overline{\mathbf{X}}}\overline{\mathcal{M}}$ if and only if for all tangent vectors $\xi_{\overline{\mathbf{X}}}, \eta_{\overline{\mathbf{X}}} \in \mathcal{T}_{\overline{\mathbf{X}}}\mathcal{M}$ the following holds

$$\langle \overline{\xi}_{\mathbf{X}_1}, \overline{\eta}_{\mathbf{X}_1} \rangle_{\mathbf{X}_1} = \langle \overline{\xi}_{\mathbf{X}_2}, \overline{\eta}_{\mathbf{X}_2} \rangle_{\mathbf{X}_2}, \; \forall\, \mathbf{X}_1, \mathbf{X}_2 \in \pi^{-1}(\overline{\mathbf{X}}).$$

Under the above assumption, the operator $\langle ., . \rangle_{\overline{\mathbf{X}}}$ on $\mathcal{T}_{\overline{\mathbf{X}}}\overline{\mathcal{M}}$ defined by $\langle \xi_{\overline{\mathbf{X}}}, \xi_{\overline{\mathbf{X}}} \rangle_{\overline{\mathbf{X}}} = \langle \overline{\xi}_{\mathbf{X}}, \overline{\eta}_{\mathbf{X}} \rangle_{\mathbf{X}}$ for any $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$ represents a well-defined Riemannian metric for the quotient manifold $\overline{\mathcal{M}}$. Let $\Pi_{\mathbf{X}}^{\mathcal{H}}$ be the orthogonal projection from the ambient space $\mathbb{R}^{n\times m}$ to the horizontal space $\mathcal{H}_{\mathbf{X}}\mathcal{M}$ and let $\mathsf{f} : \mathcal{M} \longrightarrow \mathbb{R}$ be a function that is constant on each equivalence class $[\mathbf{X}]$ for all $\mathbf{X} \in \mathcal{M}$. The above function, said to be compatible with the equivalence relationship, induces a function $\bar{\mathsf{f}} : \overline{\mathcal{M}} \longrightarrow \mathbb{R}$ such that $\bar{\mathsf{f}}(\overline{\mathbf{X}}) = \mathsf{f}(\mathbf{X})$ for any predecessor $\mathbf{X}$ of the equivalence class $\overline{\mathbf{X}}$. Under the above assumptions, the Riemannian gradient is obtained by projecting the Euclidean one onto the horizontal space of any predecessor, i.e.,

$$\mathsf{grad}\,\bar{\mathsf{f}}(\overline{\mathbf{X}}) = \Pi^{\mathcal{H}}_{\mathbf{X}}(\mathsf{Grad}\,\mathsf{f}(\mathbf{X})),\ \mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}}).$$

Second-Order Geometry of Quotient Manifolds

In the same manner as for the embedded manifold, the Riemannian Hessian can be expressed as the covariant derivative of the Riemannian gradient on the quotient manifold $\mathsf{hess}\,\bar{\mathsf{f}}(\overline{\mathbf{X}})[\xi_{\overline{\mathbf{X}}}] = \nabla_{\bar{\xi}_{\mathbf{X}}}\mathsf{grad}\,\bar{\mathsf{f}}(\overline{\mathbf{X}})$. Given that the ambient space, $\mathbb{R}^{n\times m}$ herein, is a vector space and that the Riemannian metric is induced and compatible, the connection simplifies as $\nabla_{\xi_{\overline{\mathbf{X}}}}\eta_{\overline{\mathbf{X}}} = \Pi^{\mathcal{H}}_{\mathbf{X}}(\mathsf{D}\ (\eta_{\overline{\mathbf{X}}})[\bar{\xi}_{\mathbf{X}}])$, for any $\mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}})$ which allows the Riemannian Hessian to be expressed as

$$\mathsf{hess}\,\bar{\mathsf{f}}(\overline{\mathbf{X}})[\xi_{\overline{\mathbf{X}}}] = \Pi^{\mathcal{H}}_{\mathbf{X}}(\mathsf{D}\ (\mathsf{grad}\,\bar{\mathsf{f}}(\overline{\mathbf{X}}))[\bar{\xi}_{\mathbf{X}}]).$$

Let $\overline{\mathbf{X}} \in \overline{\mathcal{M}}$ and $\mathbf{X}_1$ and $\mathbf{X}_2$ be any two *arbitrary* representatives in $\pi^{-1}(\overline{\mathbf{X}})$. Assume that the retractions $\mathsf{R}_{\mathbf{X}_1}$ and $\mathsf{R}_{\mathbf{X}_2}$ on the tangent spaces $\mathcal{T}_{\mathbf{X}_1}\mathcal{M}$ and $\mathcal{T}_{\mathbf{X}_2}\mathcal{M}$ of the manifold $\mathcal{M}$ satisfy the property $\pi(\mathsf{R}_{\mathbf{X}_1}(\bar{\xi}_{\mathbf{X}_1})) = \pi(\mathsf{R}_{\mathbf{X}_2}(\bar{\xi}_{\mathbf{X}_2}))$ for all tangent vectors. Such retraction is said to be compatible with the equivalence relationship and generate a retraction on the quotient manifold as follow

$$\mathsf{R}_{\overline{\mathbf{X}}}(\xi_{\overline{\mathbf{X}}}) = \pi(\mathsf{R}_{\mathbf{X}}(\bar{\xi}_{\mathbf{X}})), \mathbf{X} \in \pi^{-1}(\overline{\mathbf{X}}). \tag{2.4}$$

2.4 Optimization on Riemannian Manifolds

This part exploits the geometry operators introduced in the previous sections to design Riemannian optimization algorithms. In particular, the paper illustrates the a generalization of the Riemannian steepest-descent algorithm, known as line-search algorithms. Afterward, a second-order algorithm is presented in the form of the Riemannian version of Newton's method. Finally, a more-sophisticated algorithm, i.e., conjugate gradient, is introduced. Although this manuscript considers sophisticated second-order algorithms on Riemannian manifolds, such as the trust-region (TR) method, these methods can be derived using the geometry operators introduced in this manuscript and are omitted herein. Nonetheless, their performance is reported in each corresponding simulation section.

Line-Search Algorithms on Manifolds

The Riemannian version of the steepest-descent follows similar steps as the Euclidean one with the exception that the search direction is obtained with respect to the Riemannian gradient. After choosing the search direction as mandated by (2.1), the step size is selected according to Wolfe's conditions using the backtracking procedure. As stated earlier, the update step $\mathbf{X}^{t+1} = \mathbf{X}^t + \alpha^t p^t$ is trivial in Euclidean optimization thanks to its vector space structure. However, it might result in a point $\mathbf{X}^{t+1}$ outside the manifold which motivates the use of retractions. The ideal retraction is the exponential map $\text{Exp}_{\mathbf{X}}$ as it maps a tangent vector $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ to a point on the manifold along the geodesic curve (straight line on the manifold) that goes through $\mathbf{X}$ in the direction of $\xi_{\mathbf{X}}$. However, the convergence of the steepest-descent algorithm is guaranteed under the use of a first-order retraction. Therefore, the generalization of the steepest-descent algorithm to Riemannian manifolds is obtained by finding the search direction that satisfies equation (2.2) for the Riemannian metric. The update is, then, mapped to the manifold using the retraction. The steps of the method are summarized in Algorithm 2.2.

Algorithm 2.2 Template of the gradient descent procedure on Riemannian manifolds.

```
Require: Manifold $\mathcal{M}$, function $\mathsf{f}$, and retraction $\mathsf{R}$.
Initialize $\mathbf{X} \in \mathcal{M}$.
while $||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}} \geq \epsilon$ do
    Choose search direction $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ such that:
        $\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}), \xi_{\mathbf{X}} \rangle_{\mathbf{X}} < 0.$
    Compute Armijo step size $\alpha$ using backtracking.
    Retract $\mathbf{X} = \mathsf{R}_{\mathbf{X}}(\alpha\xi_{\mathbf{X}})$.
end while
Output $\mathbf{X}$.
```

Newton's Method on Riemannian Manifolds

Algorithm 2.3 Template of Newton's method on Riemannian manifolds.

```
Require: Manifold $\mathcal{M}$, function $\mathsf{f}$, retraction $\mathsf{R}$, and affine connection $\nabla$.
Initialize $\mathbf{X} \in \mathcal{M}$.
while $||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}} \geq \epsilon$ do
    Find descent direction $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ such that:
        $\mathsf{hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] = -\mathsf{grad}\ \mathsf{f}(\mathbf{X}),$
    wherein $\mathsf{hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] = \nabla_{\xi_{\mathbf{X}}}\mathsf{grad}\ \mathsf{f}(\mathbf{X})$
    Retract $\mathbf{X} = \mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}})$.
end while
Output $\mathbf{X}$.
```

Given the above definitions, the generalization of Newton's method to Riemannian manifolds is accomplished by replacing both the Euclidean gradient and Hessian by their Riemannian counterparts in (2.3). Hence, the search direction is the tangent vector $\xi_{\mathbf{X}}$ that satisfies $\mathsf{hess}\,\mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] = -\mathsf{grad}\,\mathsf{f}(\mathbf{X})$. The update is obtained by retracting the tangent vector to the manifold similar to the steepest-descent algorithm. The full steps of the algorithm are illustrated in Algorithm 2.3.

Riemannian Conjugate Gradient Algorithm

Algorithm 2.4 Template of the conjugate gradient method on Riemannian manifolds.

```
Require: Manifold $\mathcal{M}$, function $\mathsf{f}$, retraction $\mathsf{R}$, and vector transport $\mathsf{T}$.
1: Initialize $\mathbf{X} \in \mathcal{M}$.
2: Initialize $\mathbf{Y} = -\mathsf{grad}\,\mathsf{f}(\mathbf{X})$.
3: while $||\mathsf{grad}\,\mathsf{f}(\mathbf{X})||_{\mathbf{X}} \geq \epsilon$ do
4:     Compute step size $\alpha$ using line-search procedures, e.g., backtracking.
5:     Retract $\mathbf{X} = \mathsf{R}_{\mathbf{X}}(\alpha\mathbf{Y})$.
6:     Compute $\beta$ using (2.7), (2.8), or (2.9).
7:     Update $\mathbf{Y} = -\mathsf{grad}\,\mathsf{f}(\mathbf{X}) + \beta\mathsf{T}_{\alpha\mathbf{Y}}(\mathbf{Y})$.
8: end while
9: Output $\mathbf{X}$.
```

The conjugate gradient algorithm is an unconstrained optimization method developed for solving quadratic equations of the form

$$\min_{\mathbf{x}\in\mathbb{R}^n} \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x} - \mathbf{x}^{\mathrm{T}}\mathbf{b},$$

wherein matrix $\mathbf{A}$ is an $n \times n$ symmetric positive-definite matrix and $\mathbf{b} \in \mathbb{R}^n$. While simple first-order algorithms fail to converge quickly for problems with ill-conditioned matrix $\mathbf{A}$, the conjugate gradient method alleviates the problem by choosing only conjugate search directions in which the inner product is computed with respect to the matrix $\mathbf{A}$. Therefore, the main idea of conjugate gradient algorithms is to perform the update

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k, \tag{2.5}$$

wherein $\{\mathbf{p}_t\}_{t=1}^{k}$ are conjugate directions related thought the expression

$$\mathbf{p}_{k+1} = \beta_k \mathbf{p}_k - \mathsf{Grad}\,\mathsf{f}(\mathbf{x}_k), \tag{2.6}$$

with β_k can be chosen independently, yielding different nonlinear conjugate gradient methods.

The generalization of the conjugate gradient from Euclidean spaces to Riemannian manifolds is not direct. Indeed, unlike the Riemannian steepest-descent, the conjugate gradient algorithm cannot be readily extended to Riemannian manifolds as equation (2.6) combines different gradients which is not possible on manifolds. Indeed, each Riemannian gradient lives in a different tangent space. This issue is resolved throught the concept of parallel translation to connect different tangent spaces in a manifold. However, similar to the use of a retraction instead of the complicated Exponential map, one can use a vector transport instead of the more difficult to derive parallel translation. A vector transport T can be obtained by exploiting the linear structure of the embedding space and the notion of retraction as $\mathsf{T}_{\eta_{\mathbf{Y}}}(\xi_{\mathbf{Y}}) = \Pi_{\mathsf{R}_{\mathbf{Y}}(\eta_{\mathbf{Y}})}(\xi_{\mathbf{Y}})$ (see Proposition 8.1.2 [23]).

The Riemannian version of the conjugate gradient can be described as follows. Starting from an initial guess $\mathbf{X}_0$, the initial residue is computed as $\mathbf{Y}_0 = -\mathsf{grad}\ \mathsf{f}(\mathbf{X}_0)$. Afterward, while not converged, the step size α_k is computed using line-search procedures, e.g., backtracking. Generalizing the linear combination in (2.5) gives the update via the retraction $\mathbf{X}_{k+1} = \mathsf{R}_{\mathbf{X}_k}(\alpha_k \mathbf{Y}_k)$. Finally, the residue is updated as $\mathbf{Y}_{k+1} = -\mathsf{grad}\ \mathsf{f}(\mathbf{X}_{k+1}) + \beta_{k+1}\mathsf{T}_{\alpha_k \mathbf{Y}_k}(\mathbf{Y}_k)$, where the β_{k+1} can be one of the following choices

- Quasi Newton [23]

$$\beta_{k+1} = \frac{\langle \mathsf{T}_{\alpha_k \mathbf{Y}_k}(\mathbf{Y}_k), \mathsf{Hess}\ \mathsf{f}(\mathbf{X}_k)[\mathsf{grad}\ \mathsf{f}(\mathbf{X}_k)]\rangle_{\mathbf{X}_k}}{\langle \mathsf{T}_{\alpha_k \mathbf{Y}_k}(\mathbf{Y}_k), \mathsf{Hess}\ \mathsf{f}(\mathbf{X}_k)[\mathsf{T}_{\alpha_k \mathbf{Y}_k}(\mathbf{Y}_k)]\rangle_{\mathbf{X}_k}}. \tag{2.7}$$

- Fletcher-Reeves [72]:

$$\beta_{k+1} = \frac{\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}_{k+1}), \mathsf{grad}\ \mathsf{f}(\mathbf{X}_{k+1})\rangle_{\mathbf{X}_{k+1}}}{\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}_k), \mathsf{grad}\ \mathsf{f}(\mathbf{X}_k)\rangle_{\mathbf{X}_k}}. \tag{2.8}$$

- Polak-Ribiere [73]:

$$\beta_{k+1} = \frac{\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}_{k+1}), \mathsf{grad}\ \mathsf{f}(\mathbf{X}_{k+1}) - \mathsf{T}_{\alpha_k \mathbf{Y}_k}(\mathsf{grad}\ \mathsf{f}(\mathbf{X}_k)\rangle_{\mathbf{X}_{k+1}}}{\langle \mathsf{grad}\ \mathsf{f}(\mathbf{X}_k), \mathsf{grad}\ \mathsf{f}(\mathbf{X}_k)\rangle_{\mathbf{X}_k}}. \tag{2.9}$$

Finally, after obtaining the β_{k+1}, the residue is updated through the equation $\mathbf{Y}_{k+1} = -\mathsf{grad}\ \mathsf{f}(\mathbf{X}_k) + \beta_{k+1}\mathsf{T}_{\alpha \mathbf{Y}_k}(\mathbf{Y}_k)$. The steps of the algorithm are summarized in Algorithm 2.4.

Chapter 3

EFFICIENT RIEMANNIAN ALGORITHMS FOR COMMUNITY DETECTION

[1] A. Douik and B. Hassibi. “A Riemannian Approach for Graph-Based Clustering by Doubly Stochastic Matrices”. In: *Proc. of the IEEE Statistical Signal Processing Workshop (SSP’ 2018), Freiburg, Germany*. Vol. 1. 1. June 2018, pp. 806–810. doi: 10.1109/SSP.2018.8450685.

[2] A. Douik and B. Hassibi. “Manifold Optimization Over the Set of Doubly Stochastic Matrices: A Second-Order Geometry”. In: *IEEE Transactions on Signal Processing* 67.22 (Nov. 2019), pp. 5761–5774. doi: 10.1109/TSP.2019.2946024.

This chapter suggests using a Riemannian optimization approach to solve a subset of convex optimization problems wherein the optimization variable is a doubly stochastic matrix. Optimization over the set of doubly stochastic matrices is crucial for multiple communications and signal processing applications, especially graph-based clustering. The paper introduces and investigates the geometries of three convex manifolds, namely the doubly stochastic, the symmetric, and the definite multinomial manifolds which generalize the simplex, also known as the multinomial manifold. Theoretical complexity analysis and numerical simulation results testify the efficiency of the proposed method over state-of-the-art algorithms for clustering applications. In particular, they reveal that the proposed framework outperforms conventional generic and specialized approaches, especially in high dimensions. The results on this chapter appear in the research papers [74] and [75] and as such some of the text appears as it is in these publications.

3.1 Clustering via Optimization over the Set of Doubly Stochastic Matrices

State-of-the-Art Clustering Approaches

Optimization over the set of doubly stochastic matrices is a particularly interesting class of problems for its connection with probability density functions and its numerous applications in communications and signal processing, especially in graph-based clustering [39, 53, 54, 55]. Furthermore, doubly stochastic matrices play an important role in graph theory such as in critical arcs for strongly connected graphs [76] and in optimizing the mixing time of Markov chains with applications in network coding [77] and epidemics [78, 79], respectively. In addition to the aforementioned applications, the study and optimization over doubly stochastic matrices are crucial for some communication systems such as satellite-switched, time-division, multiple-access systems [80]. Finally, these matrices are also exploited in linear optimization and more specifically in the assignment problem, e.g., see [81] and references therein, which appears in various discrete optimization problems such as the resource scheduling problem.

This chapter introduces a framework for solving optimization problems in which the optimization variable is a doubly stochastic matrix. As stated earlier, such class of optimization is particularly interesting for clustering applications. In such problems, e.g., [39, 53, 54, 55], one wishes to recover the structure of a graph given a similarity or adjacency matrix. The recovery is performed by minimizing a predefined cost function over the set of doubly stochastic matrices. This work provides a unified framework to carry such optimization.

In a context of tensor decomposition, the authors in [32] propose a framework to optimize functions of stochastic matrices. This manuscript proposes extending the results to a more-general class of manifolds by introducing a framework for solving a subset of convex programs in which the optimization variable is represented by a doubly stochastic and possibly symmetric and/or definite matrix. To this end, the thesis introduces three convex manifolds which generalize the multinomial manifold. While the multinomial manifold represents stochastic matrices, the proposed manifolds characterize doubly stochastic, symmetric and definite matrices.

The paper investigates the first and second-order geometries of the proposed Riemannian manifolds endowed with the Fisher information metric which guarantees a differentiable structure. For each manifold, the tangent space, Riemannian gradient, Hessian, and retraction are derived so as to formulate first and second-order optimization algorithms. The convergence properties and theoretical complexity of these algorithms is analyzed, and simulation results are provided to further illustrate the efficiency of the proposed method against state-of-the-art algorithms.

Problems of Interest

This chapter's main contribution is to propose a framework for solving a subset of convex programs wherein the optimization variable is a doubly stochastic and possibly symmetric and/or definite matrix, i.e., a multidimensional probability distribution function. In other words, for a convex function $\mathsf{f} : \mathbb{R}^{n\times m} \to \mathbb{R}$, the paper proposes solving the following problem:

$$\begin{aligned}
&\min \mathsf{f}(\mathbf{X}) && \text{(3.1a)}\\
&\text{s}.t.\ \mathbf{X}_{ij} > 0, \forall\ 1 \leq i \leq n, 1 \leq j \leq m, && \text{(3.1b)}\\
&\qquad \sum_{j=1}^{m} \mathbf{X}_{ij} = 1, \forall\ 1 \leq i \leq n, && \text{(3.1c)}\\
&\qquad \sum_{i=1}^{n} \mathbf{X}_{ij} = 1, \forall\ 1 \leq j \leq m, && \text{(3.1d)}\\
&\qquad \mathbf{X} = \mathbf{X}^{\mathrm{T}}, && \text{(3.1e)}\\
&\qquad \mathbf{X} \succ \mathbf{0}, && \text{(3.1f)}
\end{aligned}$$

wherein constraints (3.1b)-(3.1c) produce a stochastic matrix, (3.1b)-(3.1d) a doubly stochastic one, (3.1b)-(3.1e) a symmetric stochastic one, and (3.1b)-(3.1f) a definite symmetric matrix. The study of these optimization problems is crucial as they appear in multiple communications and signal processing applications.

Manifolds and Contributions

As shown earlier, computing the Riemannian gradient and Hessian for a given function over some manifold $\mathcal{M}$ and designing a retraction allows the development of efficient first and second-order optimization algorithms that exploit the geometrical structure of the problem. Define the multinomial, doubly stochastic multinomial, symmetric multinomial, and definite multinomial, respectively, as follows:

$$\begin{aligned}
\mathcal{P}_n^m &= \left\{\mathbf{X} \in \mathbb{R}^{n\times m} \,\middle|\, \mathbf{X} > \mathbf{0},\ \mathbf{X}\mathbf{1} = \mathbf{1}\right\}\\
\mathcal{DP}_n &= \left\{\mathbf{X} \in \mathbb{R}^{n\times n} \,\middle|\, \mathbf{X} > \mathbf{0},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X}^{\mathrm{T}}\mathbf{1} = \mathbf{1}\right\}\\
\mathcal{SP}_n &= \left\{\mathbf{X} \in \mathbb{R}^{n\times n} \,\middle|\, \mathbf{X} > \mathbf{0},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X} = \mathbf{X}^{\mathrm{T}}\right\}\\
\mathcal{SP}_n^+ &= \left\{\mathbf{X} \in \mathbb{R}^{n\times n} \,\middle|\, \mathbf{X} > \mathbf{0},\ \mathbf{X}\mathbf{1} = \mathbf{1},\ \mathbf{X} = \mathbf{X}^{\mathrm{T}},\ \mathbf{X} \succ \mathbf{0}\right\},
\end{aligned}$$

wherein the notation $\mathbf{X} > \mathbf{0}$ refers to an *element-wise* positive matrix and the notation $\mathbf{X} \succ \mathbf{0}$ to a positive-definite matrix.

Recall that the notation $\mathbf{A} \oslash \mathbf{B}$ refers to the Hadamard, i.e., element-wise, division of $\mathbf{A}$ by $\mathbf{B}$. Similarly, the symbol $\odot$ denotes the Hadamard product. For all above manifolds, the paper uses the Fisher information as the Riemannian metric whose restriction to $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ is computed in [31] (Section 3.2) as follows

$$\begin{aligned}\langle \xi_{\mathbf{X}}, \eta_{\mathbf{X}} \rangle_{\mathbf{X}} &= \mathrm{Tr}((\xi_{\mathbf{X}} \oslash \mathbf{X})\eta_{\mathbf{X}}^{\mathrm{T}}) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{m} \frac{(\xi_{\mathbf{X}})_{ij}(\eta_{\mathbf{X}})_{ij}}{\mathbf{X}_{ij}}, \ \forall\, \xi_{\mathbf{X}}, \eta_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}.\end{aligned}$$

Endowing the above manifolds with the Fisher information metric ensures a differential structure that is invariant to the choice of a coordinate system. More information about the Fisher information metric and its uses in information geometry can be found in [82]. Using the definitions above, the optimization problems of interest in this chapter can be reformulated as:

$$\min_{\mathbf{X}\in\mathcal{P}_n^m} \mathsf{f}(\mathbf{X}), \qquad \min_{\mathbf{X}\in\mathcal{DP}_n} \mathsf{f}(\mathbf{X}), \qquad \min_{\mathbf{X}\in\mathcal{SP}_n} \mathsf{f}(\mathbf{X}), \qquad \min_{\mathbf{X}\in\mathcal{SP}_n^+} \mathsf{f}(\mathbf{X}).$$

3.2 The Doubly Stochastic Multinomial Manifold

Manifold Geometry

The set of doubly stochastic matrices is the set of *square* matrices with positive entries such that each column and row sums to 1. It can easily be shown that only a square matrix can verify such property. As a consequence of the Birkhoff–von Neumann theorem, $\mathcal{DP}_n$ is an embedded manifold of $\mathbb{R}^{n\times n}$. A short proof of the Birkhoff-von Neumann theorem using elementary geometry concepts can be found in [83]. The dimension of $\mathcal{DP}_n$ is $(n-1)^2$ which can be seen from the fact that the manifold is generated from $2n-1$ linearly independent equations specifying that the rows and columns all sum to one. Let $\mathbf{X} \in \mathcal{DP}_n$ be a point on the manifold, the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$ is given in the following proposition.

Proposition 3.1 *The tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$ is defined by:*

$$\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n = \left\{\xi_{\mathbf{X}} \in \mathbb{R}^{n\times n} \,\middle|\, \xi_{\mathbf{X}}\mathbf{1} = \mathbf{0},\ \xi_{\mathbf{X}}^{\mathrm{T}}\mathbf{1} = \mathbf{0}\right\}.$$

Proof: Let $\mathbf{X}(t)$ be a smooth curve such that $\mathbf{X}(0) = \mathbf{X}$. Since $\mathbf{X}(t) \in \mathcal{DP}_n$ for some t in the neighborhood of the origin, then the curve satisfies:

$$\begin{aligned}\mathbf{X}(t)\mathbf{1} = \mathbf{1} &\Rightarrow \dot{\mathbf{X}}(t)\mathbf{1} = \mathbf{0} \\ \mathbf{X}(t)^{\mathrm{T}}\mathbf{1} = \mathbf{1} &\Rightarrow \dot{\mathbf{X}}(t)^{\mathrm{T}}\mathbf{1} = \mathbf{0}.\end{aligned}$$

Differentiating both equations above concludes that the tangent space is a subset of

$$\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n \subseteq \left\{\xi_{\mathbf{X}} \in \mathbb{R}^{n\times n} \,\middle|\, \xi_{\mathbf{X}}\mathbf{1} = \mathbf{0},\ \xi_{\mathbf{X}}^{\mathrm{T}}\mathbf{1} = \mathbf{0}\right\}.$$

From the Birkhoff-von Neumann theorem [83], the degrees of freedom of the set of doubly stochastic matrices is $d = (n-1)^2$. Similarly, one can note that the above space is generated by $2n-1$ independent linear equations. Indeed, all $2n-1$ sums of columns and rows are independent except the sum of the last column which is the difference of the sum of all rows and all except the last column. In other words, the dimension of the space is $n^2 - (2n-1) = (n-1)^2$. Therefore, the set identified above has the same dimension as the tangent space which concludes that the tangent space is equal to the set above. ■

Let $\Pi_{\mathbf{X}} : \mathbb{R}^{n\times n} \longrightarrow \mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$ be the orthogonal projection from the ambient space onto the tangent space. The expression of such operator for the doubly stochastic multinomial is given below.

Theorem 3.1 *The orthogonal projection $\Pi_{\mathbf{X}}$ has the following expression:*

$$\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X},$$

wherein vectors α and β are a solution to the following over-determined but consistent $2n \times 2n$ system of equations

$$\begin{pmatrix} \mathbf{Z}\mathbf{1} \\ \mathbf{Z}^{\mathrm{T}}\mathbf{1} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \mathbf{X} \\ \mathbf{X}^{\mathrm{T}} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}.$$

Proof: In order to obtain the orthogonal projection $\Pi_{\mathbf{X}}$ onto the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$, one needs to derive a parameterization of the orthogonal complement of the tangent space $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n$ is given in the following lemma.

Lemma 3.1 *The orthogonal complement of the tangent space of the doubly stochastic multinomial has the expression:*

$$\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n = \left\{ \xi_{\mathbf{X}}^{\perp} \in \mathbb{R}^{n\times n} \;\middle|\; \xi_{\mathbf{X}}^{\perp} = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X} \right\},$$

for some vectors $\alpha, \beta \in \mathbb{R}^n$.

Proof: Let $\xi_{\mathbf{X}}^{\perp} \in \mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n$ and $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$, the inner product can be written as:

$$\begin{aligned} \langle \xi_{\mathbf{X}}^{\perp}, \xi_{\mathbf{X}} \rangle_{\mathbf{X}} &= \mathrm{Tr}((\xi_{\mathbf{X}}^{\perp} \oslash \mathbf{X})\xi_{\mathbf{X}}^{\mathrm{T}}) = \mathrm{Tr}((\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}})\xi_{\mathbf{X}}^{\mathrm{T}}) \\ &= \alpha^{\mathrm{T}}\xi_{\mathbf{X}}\mathbf{1} + \beta^{\mathrm{T}}\xi_{\mathbf{X}}^{\mathrm{T}}\mathbf{1}. \end{aligned}$$

But $\xi_{\mathbf{X}}\mathbf{1} = \xi_{\mathbf{X}}^{\mathrm{T}}\mathbf{1} = \mathbf{0}$ by definition of the tangent space. Therefore, we have $\langle \xi_{\mathbf{X}}, \xi_{\mathbf{X}}^{\perp} \rangle_{\mathbf{X}} = 0,\ \forall\ \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$. Finally, one can note that each orthogonal vector

$\xi_{\mathbf{X}}^{\perp} \in \mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n$ is completely defined by its first row and column, i.e., by $2n-1$ entries. Indeed, we have $(\xi_{\mathbf{X}}^{\perp})_{ij} = (\alpha_i + \beta_j)\mathbf{X}_{ij} = \left(\frac{(\xi_{\mathbf{X}}^{\perp})_{i1}}{\mathbf{X}_{i1}} + \frac{(\xi_{\mathbf{X}}^{\perp})_{1j}}{\mathbf{X}_{1j}} - \frac{(\xi_{\mathbf{X}}^{\perp})_{11}}{\mathbf{X}_{11}}\right)\mathbf{X}_{ij}$. Therefore, the dimension of the orthogonal complement of the tangent space is $2n-1$ which is the correct dimension for $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n$. In conclusion, the derived set is the orthogonal complement of the tangent space. ■

Let $\mathbf{Z} \in \mathbb{R}^{n\times n}$ be a vector in the ambient space and $\mathbf{X} \in \mathcal{DP}_n$. The expression of the orthogonal projection is obtained using the following decomposition:

$$\mathbf{Z} = \Pi_{\mathbf{X}}(\mathbf{Z}) + \Pi_{\mathbf{X}}^{\perp}(\mathbf{Z}) \Rightarrow \mathbf{Z1} = \Pi_{\mathbf{X}}(\mathbf{Z})\mathbf{1} + \Pi_{\mathbf{X}}^{\perp}(\mathbf{Z})\mathbf{1}. \tag{3.2}$$

However, by definition of the tangent space, the first term in the right-hand side in the above equation vanishes. Similarly, from Lemma 3.1, the second term can be replaced by its parameterization $(\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X}$. Therefore equation (3.2) implies

$$\begin{aligned}
\mathbf{Z1} &= ((\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X})\mathbf{1} \Leftrightarrow \\
\sum_{j=1}^{n} \mathbf{Z}_{ij} &= \sum_{j=1}^{n} (\alpha_i + \beta_j)\mathbf{X}_{ij}, 1 \le i \le n \Leftrightarrow \\
\sum_{j=1}^{n} \mathbf{Z}_{ij} &= \alpha_i + \sum_{j=1}^{n} \beta_j \mathbf{X}_{ij}, 1 \le i \le n \Leftrightarrow \\
\mathbf{Z1} &= \alpha + \mathbf{X}\beta.
\end{aligned} \tag{3.3}$$

A similar argument allows to conclude that $\mathbf{Z}^{\mathrm{T}}\mathbf{1} = \mathbf{X}^{\mathrm{T}}\alpha + \beta$. Grouping the equations above gives the following system

$$\begin{pmatrix} \mathbf{Z1} \\ \mathbf{Z}^{\mathrm{T}}\mathbf{1} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \mathbf{X} \\ \mathbf{X}^{\mathrm{T}} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix}. \tag{3.4}$$

By definition of the manifold, we have $\mathbf{X} > \mathbf{0}$ which translates to the fact that the Markov chain represented by $\mathbf{X}$ is irreducible. According to Perron-Frobenius theorem, $\mathbf{X}$ has a unique eigenvector at $\mathbf{1}$ associated with the eigenvalue 1. As a result, even though the matrix $\mathbf{A} = \begin{pmatrix} \mathbf{I} & \mathbf{X} \\ \mathbf{X}^{\mathrm{T}} & \mathbf{I} \end{pmatrix}$ is rank deficient as it has a null vector at $\begin{pmatrix} \mathbf{1} \\ -\mathbf{1} \end{pmatrix}$, the systems admits infinitely many solutions. Indeed, given the identification of the range space and the orthogonal complement of the null space of the matrix $\mathbf{A}$, i.e., $\mathcal{R}(\mathbf{A}) = \mathcal{N}^{\perp}(\mathbf{A})$, it is sufficient to show that the vector is orthogonal to the null space of the matrix of interest as follows:

$$\begin{aligned}\begin{pmatrix}\mathbf{Z1}\\ \mathbf{Z}^{\mathrm{T}}\mathbf{1}\end{pmatrix}^{\mathrm{T}}\begin{pmatrix}\mathbf{1}\\ -\mathbf{1}\end{pmatrix} &= \mathbf{1}^{\mathrm{T}}\mathbf{Z}^{\mathrm{T}}\mathbf{1} - \mathbf{1}^{\mathrm{T}}\mathbf{Z1}\\ &= \mathbf{1}^{\mathrm{T}}\mathbf{Z1} - \mathbf{1}^{\mathrm{T}}\mathbf{Z1} = \mathbf{0}.\end{aligned}$$

Finally, rearranging the terms in (3.2) concludes that the orthogonal projection onto the tangent space has the following expression

$$\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X},$$

wherein α and β are obtained according to (3.4). ■

Although its representation in terms of α and β is not unique, the projection $\Pi_{\mathbf{X}}(\mathbf{Z})$ is unique and does not depend on the choice of the solution to the over-determined system. The non-uniqueness of the representation follows from the fact that the $2n-1$-dimensional space $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{DP}_n$ is parametrized with $2n$ variables, namely α and β.

Riemannian Gradient and Retraction Computation

For the Fisher information metric, the Riemannian gradient $\operatorname{grad} \mathsf{f}(\mathbf{X})$ can be expressed as the projection of a scaled version of the Euclidean gradient $\operatorname{Grad} \mathsf{f}(\mathbf{X})$. Indeed, the authors in [32] demonstrate that the Riemannian gradient for the multinomial manifold is given by

$$\operatorname{grad} \mathsf{f}(\mathbf{X}) = \Pi_{\mathbf{X}}(\operatorname{Grad} \mathsf{f}(\mathbf{X}) \odot \mathbf{X}). \tag{3.5}$$

Therefore, given the expression of the orthogonal projection onto the tangent space in Theorem 3.1, the Riemannian gradient can be computed from its Euclidean counterpart using equation (3.5). Besides, note that the relationship in (3.5) depends solely on the expression of the Riemannian metric and thus applies to the three manifolds of interest in this paper.

As stated in Chap. 2, one needs to define a retraction from the tangent bundle to the manifold to take advantage of optimization algorithms on Riemannian manifolds. The canonical approach to derive an efficient retraction, e.g., a retraction that can be computed quickly, is to exploit the linear structure of the embedding space. The construction of such retraction rely on the following theorem whose proof can be found in [23].

Theorem 3.2 *Let $\mathcal{M}$ be an embedded manifold of the Euclidean space $\mathcal{E}$ and let $\mathcal{N}$ be an abstract manifold such that dim($\mathcal{M}$) + dim($\mathcal{N}$) = dim($\mathcal{E}$). Assume that there is a diffeomorphism*

$$\begin{aligned}\phi : \mathcal{M} \times \mathcal{N} &\to \mathcal{E}^\star \\ (\mathbf{F}, \mathbf{G}) &\longmapsto \phi(\mathbf{F}, \mathbf{G}),\end{aligned}$$

where $\mathcal{E}^\star$ is an open subset of $\mathcal{E}$, with a neutral element $\mathbf{I} \in \mathcal{N}$ satisfying

$$\phi(\mathbf{F}, \mathbf{I}) = \mathbf{F}, \ \forall \ \mathbf{F} \in \mathcal{M}.$$

Under the above assumptions, the map

$$\begin{aligned}\mathsf{R}_\mathbf{X} : \mathcal{T}_\mathbf{X}\mathcal{M} &\to \mathcal{M} \\ \xi_\mathbf{X} &\longmapsto \mathsf{R}_\mathbf{X}(\xi_\mathbf{X}) = \pi_1(\phi^{-1}(\mathbf{X} + \xi_\mathbf{X})),\end{aligned}$$

where $\pi_1 : \mathcal{M} \times \mathcal{N} \to \mathcal{M} : (\mathbf{F}, \mathbf{G}) \longmapsto \mathbf{F}$ is the projection onto the first component, defines a retraction on the manifold $\mathcal{M}$ for all $\mathbf{X} \in \mathcal{M}$ and $\xi_\mathbf{X}$ in the neighborhood of $\mathbf{0}_\mathbf{X}$.

Define the set of entry-wise positive matrices $\overline{\mathbb{R}}^{n\times n} = \{\mathbf{X} \in \mathbb{R}^{n\times n} \mid \mathbf{X} > \mathbf{0}\}$. The canonical retraction on the doubly stochastic multinomial manifold is given in the following theorem:

Theorem 3.3 *The map $\mathsf{R} : \mathcal{T}\mathcal{DP}_n \to \mathcal{DP}_n$ whose restriction $\mathsf{R}_\mathbf{X}$ to $\mathcal{T}_\mathbf{X}\mathcal{DP}_n$ is given by:*

$$\mathsf{R}_\mathbf{X}(\xi_\mathbf{X}) = \mathbf{X} + \xi_\mathbf{X},$$

represents a well-defined retraction on the doubly stochastic multinomial manifold provided that $\xi_\mathbf{X}$ is in the neighborhood of $\mathbf{0}_\mathbf{X}$, i.e., $\mathbf{X} + \xi_\mathbf{X} > \mathbf{0}$.

Proof: This proof uses Sinkhorn's theorem [84] to derive an expression for the map ϕ. Sinkhorn's theorem states that for an element-wise positive matrix $\mathbf{A} \in \overline{\mathbb{R}}^{n\times n}$, there exists two strictly positive diagonal matrices $\mathbf{d}_1$ and $\mathbf{d}_2$ such that $\mathbf{d}_1\mathbf{A}\mathbf{d}_2$ is doubly stochastic. Due to the invariance of the above theorem for scaling $\mathbf{d}_1$ and $\mathbf{d}_2$, the rest of the paper assumes that $(\mathbf{d}_1)_{11} = 1$ without loss of generality. Define the map ϕ as follows:

$$\begin{aligned}\phi : \mathcal{DP}_n \times \overline{\mathbb{R}}^{2n-1} &\to \overline{\mathbb{R}}^{n\times n} \\ \left(\mathbf{A}, \begin{pmatrix} d_1 \\ d_2 \end{pmatrix}\right) &\longmapsto \text{diag}(1, d_1)\mathbf{A}\text{diag}(d_2).\end{aligned}$$

Note that $\overline{\mathbb{R}}^{2n-1}$ is an open subset of $\mathbb{R}^{2n-1}$ and thus is a manifold by definition. Similarly, $\overline{\mathbb{R}}^{n\times n}$ is an open subset of $\mathbb{R}^{n\times n}$. Finally, $\dim(\mathcal{DP}_n) + \dim(\overline{\mathbb{R}}^{2n-1}) = (n-1)^2 + 2n - 1 = n^2 = \dim(\mathbb{R}^{n\times n})$. Also, the all one element of $\mathbb{R}^{2n-1}$ satisfies $\phi(\mathbf{A}, \mathbf{1}) = \mathbf{A}$. Furthermore, given that this paper is interested in points on the manifold $\mathcal{DP}_n$, then all entries of $\mathbf{A}$ are strictly positive which guarantees, according to Sinkhorn and Knopp [85], that the map ϕ is invertible.

Clearly, the map ϕ is smooth given the smoothness of the matrix product. The existence of the inverse map is guaranteed by Sinkhorn and Knopp [85] as discussed above. Such an inverse map is obtained through Sinkhorn's algorithm [84] that scales the rows and columns of the matrix. Hence, the smoothness of the inverse map derives from the fact that the map is computed as a finite sequence of smooth operations over an open set, namely the positive quadrant. Finally, we conclude that ϕ represents a diffeomorphism.

Using the result of Theorem 3.2, we obtain that $\pi_1(\phi^{-1}(\mathbf{X}+\xi_{\mathbf{X}}))$ is a valid retraction for $\xi_{\mathbf{X}}$ in the neighborhood of $\mathbf{0}_{\mathbf{X}}$, i.e., $(\mathbf{X}+\xi_{\mathbf{X}}) \in \overline{\mathbb{R}}^{n\times n}$ which can explicitly written as $\mathbf{X}_{ij} > -\left(\xi_{\mathbf{X}}\right)_{ij}$, $1 \leq i, j \leq n$. Using the properties of the manifold and its tangent space, the inverse map reduce the identity. Indeed, it holds true that:

$$(\mathbf{X}+\xi_{\mathbf{X}})\mathbf{1} = \mathbf{X}\mathbf{1} + \xi_{\mathbf{X}}\mathbf{1} = \mathbf{1} + \mathbf{0} = \mathbf{1}$$
$$(\mathbf{X}+\xi_{\mathbf{X}})^T\mathbf{1} = \mathbf{X}^T\mathbf{1} + \xi_{\mathbf{X}}^T\mathbf{1} = \mathbf{1} + \mathbf{0} = \mathbf{1}.$$

Hence, the retraction is defined by $\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathbf{X} + \xi_{\mathbf{X}}$. ■

The performance of the above retraction is satisfactory as long as the optimal solution $\mathbf{X}^\star$ does not have vanishing entries, i.e., some $\mathbf{X}^\star_{ij}$ that approaches 0. In such situation, the update procedure results in tiny steps which compromise the convergence speed of the algorithm. Although the projection on the set of doubly stochastic matrices is costly [85], this paper proposes a highly efficient retraction that takes advantage of the structure of both the manifold and its tangent space. Let $\mathsf{P} : \overline{\mathbb{R}}^{n\times n} \longrightarrow \mathcal{DP}_n$ be the map to the set of doubly stochastic matrices obtained using the Sinkhorn-Knopp algorithm [84]. The proposed retraction, using the element-wise exponential of a matrix exp(.), is given in the following lemma

Lemma 3.2 *The map* $\mathsf{R} : \mathcal{T}\mathcal{DP}_n \longrightarrow \mathcal{DP}_n$ *whose restriction* $\mathsf{R}_{\mathbf{X}}$ *to* $\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$ *is given by:*

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathsf{P}\left(\mathbf{X} \odot exp(\xi_{\mathbf{X}} \oslash \mathbf{X})\right),$$

is a first-order retraction on the doubly stochastic multinomial manifold for all $\xi_{\mathbf{X}} \in \mathcal{TDP}_n$*.*

Proof: To show that the operator represents a well-defined retraction, one needs to demonstrate that the centering and local rigidity conditions are satisfied. The centering property is straightforward, i.e.,:

$$\mathsf{R}_{\mathbf{X}}(\mathbf{0}) = \mathsf{P}\left(\mathbf{X} \odot \exp(\mathbf{0})\right) = \mathsf{P}\left(\mathbf{X}\right) = \mathbf{X},$$

wherein the last equality is obtained from the fact that $\mathbf{X}$ is already a doubly stochastic matrix.

To prove the local rigidity condition, one needs to study the behavior of $\mathsf{P}(\mathbf{X})$ around a “small" perturbation $\partial\mathbf{X}$ in the tangent space $\mathcal{TDP}_n$ wherein “small" refers to the fact that $\mathbf{X} + \partial\mathbf{X} \in \overline{\mathbb{R}}^{n\times n}$. First note from Sinkhorn-Knopp algorithm that $\mathsf{P}(\mathbf{X}) = \mathbf{d}_1\mathbf{X}\mathbf{d}_2$. However, since $\mathbf{X}$ is already doubly stochastic, then $\mathbf{d}_1 = \mathbf{d}_2 = \mathbf{I}$. The first-order approximation of the $\mathsf{P}(\mathbf{X} + \partial\mathbf{X})$ can be written as:

$$\begin{aligned}\mathsf{P}(\mathbf{X} + \partial\mathbf{X}) &= (\mathbf{d}_1 + \partial\mathbf{d}_1)(\mathbf{X} + \partial\mathbf{X})(\mathbf{d}_2 + \partial\mathbf{d}_2)\\ &= \mathbf{d}_1\mathbf{X}\mathbf{d}_2 + \mathbf{d}_1\partial\mathbf{X}\mathbf{d}_2 + \partial\mathbf{d}_1\mathbf{X}\mathbf{d}_2 + \mathbf{d}_1\mathbf{X}\partial\mathbf{d}_2\\ &= \mathbf{X} + \partial\mathbf{X} + \partial\mathbf{d}_1\mathbf{X} + \mathbf{X}\partial\mathbf{d}_2.\end{aligned}$$

Since $\mathsf{P}(\mathbf{X} + \partial\mathbf{X})$ and $\mathbf{X}$ are doubly stochastic and $\partial\mathbf{X}$ is in the tangent space, then we obtain:

$$\begin{aligned}&\mathsf{P}(\mathbf{X} + \partial\mathbf{X})\mathbf{1} = (\mathbf{X} + \partial\mathbf{X} + \partial\mathbf{d}_1\mathbf{X} + \mathbf{X}\partial\mathbf{d}_2)\mathbf{1} \Rightarrow\\ &\partial\mathbf{d}_1\mathbf{X}\mathbf{1} + \mathbf{X}\partial\mathbf{d}_2\mathbf{1} = \partial\mathbf{d}_1\mathbf{1} + \mathbf{X}\partial\mathbf{d}_2\mathbf{1} = \mathbf{0}.\end{aligned}$$

Similarly, by post multiplying by $\mathbf{1}^{\mathrm{T}}$, we obtain $\mathbf{1}^{\mathrm{T}}\partial\mathbf{d}_1\mathbf{X} + \mathbf{1}^{\mathrm{T}}\partial\mathbf{d}_2 = \mathbf{0}^{\mathrm{T}}$. For ease of notations, let $\partial\mathbf{d}_1\mathbf{1} = \partial\mathbf{d}_1$, i.e., $\partial\mathbf{d}_1$ is the vector created from the diagonal entries of $\partial\mathbf{d}_1$ and the same for $\partial\mathbf{d}_2$. Combining both equations above, the perturbation on the diagonal matrices satisfy the condition:

$$\begin{pmatrix}\mathbf{I} & \mathbf{X}\\ \mathbf{X}^{\mathrm{T}} & \mathbf{I}\end{pmatrix}\begin{pmatrix}\partial\mathbf{d}_1\\ \partial\mathbf{d}_2\end{pmatrix} = \begin{pmatrix}\mathbf{0}\\ \mathbf{0}\end{pmatrix}.$$

In other words, $\begin{pmatrix}\partial\mathbf{d}_1\\ \partial\mathbf{d}_2\end{pmatrix}$ is in the null space of the above matrix which is generated by $\begin{pmatrix}\mathbf{1}\\ -\mathbf{1}\end{pmatrix}$ from the previous analysis. As a result, $\partial\mathbf{d}_1 = -\partial\mathbf{d}_2 = c\mathbf{1}$ which gives

$\partial \mathbf{d}_1 \mathbf{X} + \mathbf{X} \partial \mathbf{d}_2 = \mathbf{0}$. Therefore, $\mathsf{P}(\mathbf{X} + \partial \mathbf{X}) = \mathbf{X} + \partial \mathbf{X} + o(||\partial \mathbf{X}||^2)$. Now, consider the curve $\gamma_{\xi_\mathbf{X}}(\tau) = \mathsf{R}_\mathbf{X}(\tau \xi_\mathbf{X})$. The derivative of the curve at the origin can be written as:

$$\left.\frac{d\gamma_{\xi_\mathbf{X}}(\tau)}{d\tau}\right|_{\tau=0} = \lim_{\tau \to 0} \frac{\mathsf{P}(\mathbf{X} \odot \exp(\tau \xi_\mathbf{X} \oslash \mathbf{X})) - \mathbf{X}}{\tau}. \tag{3.6}$$

A first-order approximation of the exponential allows to express the first term in the denominator as:

$$\mathsf{P}(\mathbf{X} \odot \exp(\tau \xi_\mathbf{X} \oslash \mathbf{X})) = \mathsf{P}(\mathbf{X} + \tau \xi_\mathbf{X}) = \mathbf{X} + \tau \xi_\mathbf{X}.$$

wherein the last equality is obtained from the previous analysis. Plugging the expression in the limit expression shows the local rigidity condition. ■

Connection and Riemannian Hessian Computation

The computation of the Riemannian Hessian requires the derivation of the Levi-Civita connection $\nabla_\eta \xi$. Using the result of [23], the Levi-Civita connection of a submanifold $\mathcal{M}$ of the Euclidean space $\mathbb{R}^{n \times n}$ can be obtained by projecting the Levi-Civita $\overline{\nabla}_{\eta_\mathbf{X}} \xi_\mathbf{X}$ of the embedding space onto the manifold, i.e., $\nabla_{\eta_\mathbf{X}} \xi_\mathbf{X} = \Pi_\mathbf{X}(\overline{\nabla}_{\eta_\mathbf{X}} \xi_\mathbf{X})$. From Koszul's formula, the connection $\overline{\nabla}_{\eta_\mathbf{X}} \xi_\mathbf{X}$ on $\mathbb{R}^{n \times n}$ solely depends on the Riemannian metric. In other words, the connection $\overline{\nabla}_{\eta_\mathbf{X}} \xi_\mathbf{X}$ on the embedding space is the same for all manifolds in this paper. For manifolds endowed with the Fisher information metric, the Levi-Civita connection on $\mathbb{R}^{n \times n}$ is given in [32] as

$$\overline{\nabla}_{\eta_\mathbf{X}} \xi_\mathbf{X} = \mathrm{D}(\xi_\mathbf{X})[\eta_\mathbf{X}] - \frac{1}{2}(\eta_\mathbf{X} \odot \xi_\mathbf{X}) \oslash \mathbf{X}. \tag{3.7}$$

Therefore, the Riemannian Hessian can be written as a function of the Euclidean gradient and Hessian as follows:

Theorem 3.4 *The Riemannian Hessian* $\mathsf{hess}\, \mathsf{f}(\mathbf{X})[\xi_\mathbf{X}]$ *can be obtained from the Euclidean gradient* $\mathsf{Grad}\, \mathsf{f}(\mathbf{X})$ *and the Euclidean Hessian* $\mathsf{Hess}\, \mathsf{f}(\mathbf{X})[\xi_\mathbf{X}]$ *using the* *identity:*

$$\begin{aligned}
\text{hess f}(\mathbf{X})[\xi_{\mathbf{X}}] &= \Pi_{\mathbf{X}}\left(\dot{\delta} - \frac{1}{2}(\delta \odot \xi_{\mathbf{X}}) \oslash \mathbf{X}\right) \\
\alpha &= \epsilon(\gamma - \mathbf{X}\gamma^{\mathrm{T}})\mathbf{1} \\
\beta &= \gamma^{\mathrm{T}}\mathbf{1} - \mathbf{X}^{T}\alpha \\
\gamma &= \text{Grad f}(\mathbf{X}) \odot \mathbf{X} \\
\delta &= \gamma - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \mathbf{X} \\
\epsilon &= (\mathbf{I} - \mathbf{X}\mathbf{X}^{\mathrm{T}})^{\dagger} \\
\dot{\alpha} &= \left[\dot{\epsilon}(\gamma - \mathbf{X}\gamma^{\mathrm{T}}) + \epsilon(\dot{\gamma} - \xi_{\mathbf{X}}\gamma - \mathbf{X}\dot{\gamma}^{\mathrm{T}})\right]\mathbf{1} \\
\dot{\beta} &= \dot{\gamma}^{\mathrm{T}}\mathbf{1} - \xi_{\mathbf{X}}^{\mathrm{T}}\alpha - \mathbf{X}^{T}\dot{\alpha} \\
\dot{\gamma} &= \text{Hess f}(\mathbf{X})[\xi_{\mathbf{X}}] \odot \mathbf{X} + \text{Grad f}(\mathbf{X}) \odot \xi_{\mathbf{X}} \\
\dot{\delta} &= \dot{\gamma} - (\dot{\alpha}\mathbf{1}^{\mathrm{T}} + \mathbf{1}\dot{\beta}^{\mathrm{T}}) \odot \mathbf{X} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}}) \odot \xi_{\mathbf{X}} \\
\dot{\epsilon} &= \epsilon(\mathbf{X}\xi_{\mathbf{X}}^{\mathrm{T}} + \xi_{\mathbf{X}}\mathbf{X}^{\mathrm{T}})\epsilon.
\end{aligned}$$

Proof: Recall that the Riemannian Hessian is related to the Riemannian connection and Riemannian gradient through the following equation:

$$\begin{aligned}
\text{hess f}(\mathbf{X})[\xi_{\mathbf{X}}] &= \nabla_{\xi_{\mathbf{X}}}\text{grad f}(\mathbf{X}), \ \forall\, \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}. \\
&= \Pi_{\mathbf{X}}(\overline{\nabla}_{\xi_{\mathbf{X}}}\text{grad f}(\mathbf{X})),
\end{aligned}$$

with the Levi-Civita connection $\overline{\nabla}_{\eta_{\mathbf{X}}}\xi_{\mathbf{X}}$, i.e., covariant derivative, being defined in (3.7). Therefore, the computation of the Hessian only requires computing the covariant derivative of the Riemannian gradient and projecting it onto the tangent space which is omitted herein for conciseness. ■

3.3 The Symmetric Multinomial Manifold

This section derives the first and second-order geometries of the symmetric and the definite multinomial manifolds. As the geometries of both manifolds are alike, the two first parts derive the expression of the tangent space, Riemannian gradient and Hessian. Finally, the final part provides an expression of a retraction that exploits the structure of the definite multinomial manifold.

Whereas the doubly stochastic multinomial manifold is regarded as an embedded manifold of the space of matrices $\mathbb{R}^{n\times n}$, the symmetric and definite multinomial manifolds are seen as embedded manifolds of the set of symmetric matrices. In other words, the embedding Euclidean space is the space of symmetric matrices $\mathcal{S}_n$ defined as:

$$\mathcal{S}_n = \left\{ \mathbf{X} \in \mathbb{R}^{n \times n} \,\middle|\, \mathbf{X} = \mathbf{X}^{\mathrm{T}} \right\}.$$

Such choice of ambient space allows to reduce the ambient dimension from n^2 to $\frac{n(n+1)}{2}$ and thus enables the simplification of the projection operator. As a result, the Riemannian gradient and Hessian expressions can be computed more efficiently.

Manifold Geometry, Gradient, and Retraction

Let $\mathbf{X} \in \mathcal{SP}_n$ be a point on the manifold, the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ is given in the following proposition.

Proposition 3.2 *The tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ is defined by:*

$$\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n = \left\{ \xi_{\mathbf{X}} \in \mathcal{S}_n \,\middle|\, \xi_{\mathbf{X}}\mathbf{1} = \mathbf{0} \right\}.$$

Proof: A smooth curve $\mathbf{X}(t)$ that goes through a point $\mathbf{X} \in \mathcal{SP}_n$ satisfies:

$$\mathbf{X}(t) = \mathbf{X}(t)^{\mathrm{T}} \Rightarrow \dot{\mathbf{X}}(t) = \dot{\mathbf{X}}(t)^{\mathrm{T}}$$
$$\mathbf{X}(t)\mathbf{1} = \mathbf{1} \Rightarrow \dot{\mathbf{X}}(t)\mathbf{1} = \mathbf{0},$$

which concludes that the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{DP}_n$ is included in the set $\{\xi_{\mathbf{X}} \in \mathcal{S}_n \,|\, \xi_{\mathbf{X}}\mathbf{1} = \mathbf{0}\}$. Now consider $\xi_{\mathbf{X}}$ in the above set and the smooth curve $\gamma(t) = \mathbf{X} + t\xi_{\mathbf{X}}$. Clearly, $\gamma(t) = \gamma(t)^{\mathrm{T}}$ for all $t \in \mathbb{R}$. Furthermore, we have:

$$\gamma(t)\mathbf{1} = \mathbf{X}\mathbf{1} + t\xi_{\mathbf{X}}\mathbf{1} = \mathbf{X}\mathbf{1} = \mathbf{1}.$$

Finally, since $\mathbf{X} > \mathbf{0}$ defines an open set, there exists an interval $\mathcal{I} \subset \mathbb{R}$ such that $\gamma(t) > \mathbf{0}$. Finally, it is clear that $\gamma(\mathbf{0}) = \mathbf{X}$ and $\gamma'(\mathbf{0}) = \xi_{\mathbf{X}}$ which concludes that:

$$\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n = \left\{ \xi_{\mathbf{X}} \in \mathcal{S}_n \,\middle|\, \xi_{\mathbf{X}}\mathbf{1} = \mathbf{0} \right\}.$$

■

Let $\Pi_{\mathbf{X}} : \mathcal{S}_n \longrightarrow \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ be the orthogonal projection from the ambient space onto the tangent space. Note that the ambient space for the symmetric multinomial $\mathcal{SP}_n$ is the set of symmetric matrices $\mathcal{S}_n$ and not the set of all matrices $\mathbb{R}^{n \times n}$. The following theorem gives the expression of the projection operator:

Theorem 3.5 *The orthogonal projection* $\Pi_{\mathbf{X}}$ *onto the tangent space* $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ *has the following expression*

$$\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X},$$

wherein the vector α *is computed as:*

$$\alpha = (\mathbf{I} + \mathbf{X})^{-1}\mathbf{Z}\mathbf{1}.$$

Proof: Using similar steps as in Lemma 3.1, the expression of the orthogonal complement of the tangent space is given by the following set:

$$\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{SP}_n = \left\{\xi_{\mathbf{X}}^{\perp} \in \mathcal{S}_n \mid \xi_{\mathbf{X}}^{\perp} = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X}\right\},$$

for some $\alpha \in \mathbb{R}^n$. Let $\mathbf{Z} \in \mathcal{S}_n$ be a vector in the ambient space and $\mathbf{X} \in \mathcal{DP}_n$. The decomposition of $\mathbf{Z}$ gives

$$\begin{aligned} \mathbf{Z} &= \Pi_{\mathbf{X}}(\mathbf{Z}) + \Pi_{\mathbf{X}}^{\perp}(\mathbf{Z}) \Rightarrow \\ \mathbf{Z}\mathbf{1} &= \Pi_{\mathbf{X}}(\mathbf{Z})\mathbf{1} + \Pi_{\mathbf{X}}^{\perp}(\mathbf{Z})\mathbf{1} \Rightarrow \\ \mathbf{Z}\mathbf{1} &= ((\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X})\mathbf{1} \Rightarrow \\ \mathbf{Z}\mathbf{1} &= \alpha + \mathbf{X}\alpha = (\mathbf{I} + \mathbf{X})\alpha \Rightarrow \\ \alpha &= (\mathbf{I} + \mathbf{X})^{-1}\mathbf{Z}\mathbf{1}, \end{aligned} \tag{3.8}$$

wherein the steps of the computation are obtained in a similar fashion as in (3.3). Therefore, the orthogonal projection on the tangent space of the symmetric multinomial manifold is given by:

$$\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X},$$

with α being derived in (3.8). ■

Given the expression of the projection onto the tangent space, the Riemannian gradient can be efficiently computed as:

$$\begin{aligned} \text{grad } \mathsf{f}(\mathbf{X}) &= \gamma - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X} \\ \alpha &= (\mathbf{I} + \mathbf{X})^{-1}\gamma\mathbf{1} \\ \gamma &= (\text{Grad } \mathsf{f}(\mathbf{X}) \odot \mathbf{X}). \end{aligned}$$

Similar to the doubly stochastic multinomial manifold, the canonical retraction on the symmetric multinomial manifold can be efficiently computed as shown in the following corollary.

Corollary 3.1 *The map* $\mathsf{R} : \mathcal{T}\mathcal{SP}_n \longrightarrow \mathcal{SP}_n$ *whose restriction* $\mathsf{R}_{\mathbf{X}}$ *to* $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ *is given by:*

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathbf{X} + \xi_{\mathbf{X}},$$

represents a well-defined retraction on the symmetric multinomial manifold provided that $\xi_{\mathbf{X}}$ *is in the neighborhood of* $\mathbf{0}_{\mathbf{X}}$*, i.e.,* $\mathbf{X} + \xi_{\mathbf{X}} > \mathbf{0}$.

The proof of this corollary is omitted herein as it follows similar steps than Theorem 3.3. However, instead of using Sinkhorn's theorem to find the adequate matrix decomposition, we use its extension to symmetric matrices known as the DAD theorem [86].

The canonical retraction suffers from the same limitations discussed in the previous section. Indeed, the performance of the optimization algorithm heavily depend on whether the optimal solution has vanishing entries or not. This section shows that the retraction proposed in Lemma 3.2 is a valid retraction on the set of symmetric double stochastic matrices. However, instead of the Sinkhorn-Knopp algorithm [84], this part uses the DAD algorithm [86] to project the retracted vector. Let $\overline{\mathcal{S}}_n = \left\{\mathbf{X} \in \mathbb{R}^{n \times n} \,\middle|\, \mathbf{X} > \mathbf{0},\ \mathbf{X} = \mathbf{X}^{\mathrm{T}}\right\}$ represent the set of symmetric, element-wise positive matrices. The map to the set of symmetric doubly stochastic matrices is denoted by the operator $\mathsf{P}^{+} : \overline{\mathcal{S}}_n \longrightarrow \mathcal{SP}_n$. The retraction is given in the following corollary whose proof is omitted as it mirrors the steps used in the proof of Lemma 3.2.

Corollary 3.2 *The map* $\mathsf{R} : \mathcal{T}\mathcal{SP}_n \longrightarrow \mathcal{SP}_n$ *whose restriction* $\mathsf{R}_{\mathbf{X}}$ *to* $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n$ *is given by:*

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathsf{P}^{+}\left(\mathbf{X} \odot exp(\xi_{\mathbf{X}} \oslash \mathbf{X})\right),$$

is a first-order retraction on the symmetric multinomial manifold for all $\xi_{\mathbf{X}} \in \mathcal{T}\mathcal{SP}_n$.

Connection and Riemannian Hessian Computation

As discussed earlier, the Levi-Civita connection solely depends on the Riemannian metric. Therefore, the symmetric multinomial manifold shares the same connection on the embedding space as the doubly stochastic multinomial manifold. The Riemannian Hessian can be obtained by differentiating the Riemannian gradient using the projection of the Levi-Civita connection onto the manifold as shown in the below corollary:

Corollary 3.3 *The Riemannian Hessian* $\mathsf{hess}\,\mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$ *can be obtained from the Euclidean gradient* $\mathsf{Grad}\,\mathsf{f}(\mathbf{X})$ *and the Euclidean Hessian* $\mathsf{Hess}\,\mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}]$ *using the identity:*

$$\begin{aligned}
\mathsf{hess}\,\mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] &= \Pi_{\mathbf{X}}\left(\dot{\delta} - \frac{1}{2}(\delta \odot \xi_{\mathbf{X}}) \oslash \mathbf{X}\right)\\
\alpha &= (\mathbf{I}+\mathbf{X})^{-1}\gamma\mathbf{1}\\
\delta &= \gamma - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \mathbf{X}\\
\gamma &= \mathsf{Grad}\,\mathsf{f}(\mathbf{X}) \odot \mathbf{X}\\
\dot{\alpha} &= \left((\mathbf{I}+\mathbf{X})^{-1}\dot{\gamma} - (\mathbf{I}+\mathbf{X})^{-1}\xi_{\mathbf{X}}(\mathbf{I}+\mathbf{X})^{-1}\gamma\right)\mathbf{1}\\
\dot{\delta} &= \dot{\gamma} - (\dot{\alpha}\mathbf{1}^{\mathrm{T}} + \mathbf{1}\dot{\alpha}^{\mathrm{T}}) \odot \mathbf{X} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}}) \odot \xi_{\mathbf{X}}\\
\dot{\gamma} &= \mathsf{Hess}\,\mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] \odot \mathbf{X} + \mathsf{Grad}\,\mathsf{f}(\mathbf{X}) \odot \xi_{\mathbf{X}}.
\end{aligned}$$

Proof: The proof of this corollary follows similar steps like those of Theorem 3.4 and thus is omitted herein. ■

Extension to the Definite Multinomial Manifold

The geometry of the definite multinomial manifold is similar to the symmetric multinomial manifold and thus one can use the machinery developed above to design optimization algorithms on the definite multinomial manifold. However, even though the proposed retraction is valid, its implementation is more problematic as it includes a condition on the eigenvalues. Indeed, the positive-definiteness constraint is a difficult one to retract. In order to produce highly efficient algorithms, one usually needs a re-parameterization of the positive-definiteness condition of the manifold and to regard the new structure as a quotient manifold, e.g., a Grassmann manifold. However, this falls outside the scope of this chapter and is left for low-rank matrices in the next chapter. Nonetheless, this subsection proposes another retraction that exploits the definite structure of the manifold and uses the matrix exponential to retract tangent vectors as shown in the following theorem.

Theorem 3.6 *Define the map* $\mathsf{R}_{\mathbf{X}}$ *from* $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$ *to* $\mathcal{SP}_n^+$ *by:*

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathbf{X} + \frac{1}{\omega_{\mathbf{X}}}\mathbf{I} - \frac{1}{\omega_{\mathbf{X}}}\mathbf{e}^{-\omega_{\mathbf{X}}\xi_{\mathbf{X}}},$$

wherein $\mathbf{e}^{\mathbf{X}}$ the matrix exponential of matrix $\mathbf{X}$ and $\omega_{\mathbf{X}}$ is a scalar that ensures:

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) > \mathbf{0}, \tag{3.9}$$

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) \succ \mathbf{0}, \tag{3.10}$$

for all $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$ in the neighborhood of $\mathbf{0}_{\mathbf{X}}$, i.e., $||\xi_{\mathbf{X}}||_F^2 \leq \epsilon$ for some $\epsilon > 0$. Then, the map $\mathsf{R} : \mathcal{TSP}_n^+ \longrightarrow \mathcal{SP}_n^+$, whose restriction $\mathsf{R}_{\mathbf{X}}$ to $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$, is a retraction on the definite multinomial manifold.

Proof: Unlike the canonical retraction that relies on the Euclidean structure of the embedding space, this retraction is obtained by direct computation of the properties given in Chap. 2. The organization of the proof is the following: First, assuming the existence of $\omega_{\mathbf{X}}$, we show that the range of the map $\mathsf{R}_{\mathbf{X}}$ is included in the definite symmetric manifold. Afterward, we demonstrate that the operator satisfies the centering and local rigidity conditions. Therefore, the operator represents a retraction. Finally, showing the existence of the scalar $\omega_{\mathbf{X}}$ for an arbitrary $\mathbf{X} \in \mathcal{SP}_n^+$ concludes the proof.

Recall that the matrix exponential of a symmetric real matrix $\xi_{\mathbf{X}}$ with eigenvalue decomposition $\xi_{\mathbf{X}} = \mathbf{U}\Lambda\mathbf{U}^T$ is given by $\mathbf{e}^{\xi_{\mathbf{X}}} = \mathbf{U}\exp(\Lambda)\mathbf{U}^T$, where $\exp(\Lambda)$ is the usual element-wise exponential of the element on the diagonal and zeros elsewhere. From the derivation of the tangent space of the definite symmetric multinomial manifold $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$, we have $\xi_{\mathbf{X}}\mathbf{1} = \mathbf{0}$. Therefore, $\xi_{\mathbf{X}}$ has an eigenvalue of 0 corresponding to the eigenvector $\mathbf{1}$. One can see from the definition of the matrix exponential that the eigenvalues are exponentiated while the eigenvectors are unchanged. Therefore, $\mathbf{e}^{\xi_{\mathbf{X}}}$ (and thus $\mathbf{e}^{-\omega_{\mathbf{X}}\xi_{\mathbf{X}}}$) has an eigenvalue $\exp(0) = 1$ corresponding to the eigenvector $\mathbf{1}$, i.e., $\mathbf{e}^{-\omega_{\mathbf{X}}\xi_{\mathbf{X}}}\mathbf{1} = \mathbf{1}$. Now, computing the rows summation gives:

$$\begin{aligned}\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}})\mathbf{1} &= \mathbf{X}\mathbf{1} + \frac{1}{\omega_{\mathbf{X}}}\mathbf{I}\mathbf{1} - \frac{1}{\omega_{\mathbf{X}}}\mathbf{e}^{-\omega_{\mathbf{X}}\xi_{\mathbf{X}}}\mathbf{1}\\ &= \mathbf{1} + \frac{1}{\omega_{\mathbf{X}}}\mathbf{1} - \frac{1}{\omega_{\mathbf{X}}}\mathbf{1} = \mathbf{1}.\end{aligned}$$

Hence $\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}})$ is stochastic. Besides, all matrices in the expression of $\mathsf{R}_{\mathbf{X}}$ are symmetric which concludes that the matrix is doubly stochastic. Finally, the conditions on $\omega_{\mathbf{X}}$ ensure the element-wise positiveness and the definiteness of the matrix which concludes that $\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) \in \mathcal{SP}_n^+$.

The centering property can be easily checked by evaluating the retraction $\mathsf{R}_{\mathbf{X}}$ at the zero-element $\mathbf{0}_{\mathbf{X}}$ of $\mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$. Indeed, we obtain:

$$\begin{aligned}\mathsf{R}_{\mathbf{X}}(\mathbf{0}_{\mathbf{X}}) &= \mathbf{X} + \frac{1}{\omega_{\mathbf{X}}}\mathbf{I} - \frac{1}{\omega_{\mathbf{X}}}\mathbf{e}^{-\omega_{\mathbf{X}}\mathbf{0}_{\mathbf{X}}} \\ &= \mathbf{X} + \frac{1}{\omega_{\mathbf{X}}}\mathbf{I} - \frac{1}{\omega_{\mathbf{X}}}\mathbf{I} = \mathbf{X}.\end{aligned}$$

The speed of the rigidity curve $\gamma_{\xi_{\mathbf{X}}}(\tau) = \mathsf{R}_{\mathbf{X}}(\tau\xi_{\mathbf{X}})$ at the origin is given by:

$$\frac{d\gamma_{\xi_x}(\tau)}{d\tau}\Big|_{\tau=0} = -\frac{1}{\omega_{\mathbf{X}}}\frac{d\mathbf{e}^{-\omega_{\mathbf{X}}\tau\xi_{\mathbf{X}}}}{d\tau}\Big|_{\tau=0} = \xi_{\mathbf{X}}.$$

Therefore, we conclude that $\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}})$ is a well-defined retraction.

The existence of the weight $\omega_{\mathbf{X}}$ is ensured by the fact that $\overline{\mathcal{S}}_n^+$ is an open subset of $\mathcal{S}_n$. Indeed, consider a positive sequence $\{\omega_{\mathbf{X}}^m\}_{m=1}^{\infty}$ decreasing to 0 and construct the function series $\{\mathbf{X}_m\}_{m=1}^{\infty}$ as follows:

$$\mathbf{X}_m(\xi_{\mathbf{X}}) = \mathbf{X} + \frac{1}{\omega_{\mathbf{X}}^m}\mathbf{I} - \frac{1}{\omega_{\mathbf{X}}^m}\mathbf{e}^{-\omega_{\mathbf{X}}^m\xi_{\mathbf{X}}}.$$

We aim to show that $\{\mathbf{X}_m\}_{m=1}^{\infty}$ uniformly converges to the constant $\mathbf{X} \in \overline{\mathcal{S}}_n^+$. Since $\overline{\mathcal{S}}_n^+$ is an open set, then there exists an index m_0 above which, i.e., $\forall\ m \geq m_0$ the sequence $\mathbf{X}_m(\xi_{\mathbf{X}}) \in \overline{\mathcal{S}}_n^+, \forall\ \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$ with $||\xi_{\mathbf{X}}||_F \leq \epsilon$. Hence $\omega_{\mathbf{X}}$ can be chosen to be any $\omega_{\mathbf{X}}^m$ for $m \geq m_0$.

The uniform convergence of the function series $\{\mathbf{X}_m\}_{m=1}^{\infty}$ is satisfied as $\forall\ \epsilon' > 0$, $\exists\ M_0$ such that $\forall\ m \geq M_0$ the following holds:

$$||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F < \epsilon',\ \forall\ \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+ \text{ with } ||\xi_{\mathbf{X}}||_F \leq \epsilon$$

Indeed, note that the condition over all tangent vectors can be replaced by the following condition (up to an abuse of notation with the ϵ'):

$$\begin{aligned}&||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F < \epsilon',\ \forall\ \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+ \text{ with } ||\xi_{\mathbf{X}}||_F \leq \epsilon \\ &\quad\Leftrightarrow \sup_{\substack{\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+ \\ ||\xi_{\mathbf{X}}||_F \leq \epsilon}} ||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F^2 < \epsilon' \\ &\qquad\Leftrightarrow \max_{\substack{\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+ \\ ||\xi_{\mathbf{X}}||_F \leq \epsilon}} ||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F^2 < \epsilon',\end{aligned} \tag{3.11}$$

wherein the last equivalence is obtained from the fact that the search space is closed. The last expression allows us to work with an upper bound of the distance. Indeed,

Table 3.1: Complexity of the steepest-descent algorithm and Newton's method.

Manifold	Steepest-descent algorithm	Newton's method
$\mathcal{DP}_n$	$(16/3)n^3 + 7n^2 + \log(n)\sqrt{n}$	$32/3n^3 + 15n^2 + \log(n)\sqrt{n}$
$\mathcal{SP}_n$	$(1/3)n^3 + 2n^2 + 2n + \log(n)\sqrt{n}$	$n^3 + 8n^2 + 17/2n + \log(n)\sqrt{n}$
$\mathcal{SP}_n^+$	$n^3 + 3n^2 + 3n$	$4/3n^3 + 13/2n^2 + 7n$

the distance can be bound by:

$$\begin{aligned} ||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F^2 &= \frac{1}{(\omega_{\mathbf{X}}^m)^2}||\mathbf{I} - \mathbf{e}^{-\omega_{\mathbf{X}}^m \xi_{\mathbf{X}}}||_F^2 \\ &= \frac{1}{(\omega_{\mathbf{X}}^m)^2}\sum_{i=1}^{n}(1 - \exp\Big(-\omega_{\mathbf{X}}^m \lambda_i\Big))^2 \\ &\leq \frac{n}{(\omega_{\mathbf{X}}^m)^2}(1 - \exp\Big(-\omega_{\mathbf{X}}^m \epsilon\Big))^2, \end{aligned} \tag{3.12}$$

with the last inequality being obtained from $||\xi_{\mathbf{X}}||_F \leq \epsilon \Rightarrow \lambda_i \leq \epsilon, 1 \leq i \leq n$. Now using the fact that $\{\omega_{\mathbf{X}}^m\}_{m=1}^{\infty}$ is decreasing to 0, then there exists M_0 such that $\forall\ m \geq M_0$, the following is true:

$$\frac{n}{(\omega_{\mathbf{X}}^m)^2}(1 - \exp\Big(-\omega_{\mathbf{X}}^m \epsilon\Big))^2 \leq \epsilon \leq \epsilon' \tag{3.13}$$

Combining the above results, we find out that $\forall\ \epsilon' > 0$, $\exists\ M_0$ such that $\forall\ m \geq M_0$ the following holds:

$$||\mathbf{X}_m(\xi_{\mathbf{X}}) - \mathbf{X}||_F < \epsilon',\ \forall\ \xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+ \text{ with } ||\xi_{\mathbf{X}}||_F \leq \epsilon$$

Finally, as stated earlier, combining the uniform convergence and the fact that $\overline{\mathcal{S}}_n^+$ is an open subset of $\mathcal{S}_n$ allows us to conclude the existence of $\omega_{\mathbf{X}}$ such that both conditions (3.9) and (3.10) are satisfied for all tangent vectors $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{SP}_n^+$ with $||\xi_{\mathbf{X}}||_F \leq \epsilon$. ■

3.4 Convergence and Theoretical Complexity

Convergence of the Proposed Methods

All proposed manifolds in this chapter are open convex manifolds. Therefore, given a convex objective function, the resulting Riemannian optimization problem is convex. As a result, local first and second-order Riemannian methods are globally convergent to the global minimum of the objective function [23]. Mirroring their Euclidean counterparts, the Riemannian steepest-descent algorithm and Riemannian Newton's methods have a linear and a superlinear (at least quadratic) rate of convergence to the optimum. For general objective functions, these local methods converge to an extreme point of the problem which, in practice, is very likely to be a local minimum unless the initial point is carefully crafted [23]. The rest of this section analyses the complexity of the proposed steepest-descent algorithm and Newton's method. These complexities are summarized for each manifold in Table 3.1.

Riemannian Gradient Descent Complexity

The complexity of computing the gradient for the doubly stochastic multinomial manifold can be decomposed into the complexity of computing γ, α, and $\mathsf{grad}\, \mathsf{f}(\mathbf{X})$. The term γ is a simple Hamadard product that can be computed in n^2 operations. The term α is obtained by solving a system of equations which takes $(2/3)(2n)^3$ when using an LU factorization. Finally, the expression of $\mathsf{grad}\, \mathsf{f}(\mathbf{X})$ requires a couple of additions and an Hadamard product which can be done in $3n^2$ operations. Finally, the complexity of computing the retraction can be decomposed into the complexity of computing the update vector and the complexity of the projection. The update vector requires an Hadamard product and division that can be computed in at most $3n^2$. The complexity of projecting a matrix $\mathbf{A}$ onto the set of doubly stochastic manifold [87] with accuracy ϵ is given by:

$$\mathcal{O}((1/\epsilon + \log(n))\sqrt{n}V/v), \tag{3.14}$$

wherein $V = \max(\mathbf{A})$ and $v = \min(\mathbf{A})$. Therefore, the total complexity of an iteration of the gradient descent algorithm on the doubly stochastic multinomial manifold is $(16/3)n^3 + 7n^2 + \log(n)\sqrt{n}$.

The complexity of the symmetric multinomial manifold can be obtained in a similar manner. Due to symmetry, the term γ only requires $n(n+1)/2$ operations. The term α is the solution to an $n \times n$ system of equations which can be solved in $(1/3)n^3$. Similarly, $\mathsf{grad}\, \mathsf{f}(\mathbf{X})$ requires $3/2n(n+1)$ operations. Therefore, the total complexity can be written as:

$$(1/3)n^3 + 2n^2 + 2n + \log(n)\sqrt{n}. \tag{3.15}$$

The retraction on the cone of positive matrices requires $n^3 + 2n^2$ which gives a total complexity of the algorithm for the definite multinomial manifold of $n^3 + 3n^2 + 3n$.

Riemannian Newton's Method Complexity

Computing Newton's method involves computing the Riemannian gradient and Hessian and solving an $n \times n$ linear system of equations. However, from the expression of the Riemannian Hessian, one can note that the complexity of computing the Riemannian gradient is included in the Riemannian Hessian.

For the doubly stochastic manifold, the complexity of computing the Riemannian Hessian is controlled by the complexity of the projection and the inversions. The projection onto the tangent space requires solving a $n \times n$ system of equations and a couple of additions and an Hadamard product. The total cost of the operation is $2/3(2n)^3 + 3n^2$. The ϵ and $\dot{\epsilon}$ terms are inversions and matrices products that require $4n^3$. The other terms combined require $9n^2$ operations. The retraction costs $3n^2 + \log(n)\sqrt{n}$ and solving for the search direction requires $2/3(2n)^3$ which gives a total complexity of:

$$32/3n^3 + 15n^2 + \log(n)\sqrt{n}. \tag{3.16}$$

A similar analysis as the one above allows to conclude that the total complexity of a second-order method on the symmetric and positive doubly stochastic manifold require, respectively, the following number of iterations:

$$n^3 + 8n^2 + 17/2n + \log(n)\sqrt{n} \tag{3.17}$$

$$4/3n^3 + 13/2n^2 + 7n. \tag{3.18}$$

3.5 Simulation Results

This section attests the performance of the proposed framework in efficiently solving optimization problems in which the optimization variable is a doubly stochastic matrix. The experiments are carried out using MATLAB on an Intel Xeon Processor E5-1650 v4 (15M Cache, 3.60 GHz) computer with 32GB 2.4 GHz DDR4 RAM. The steepest-descent algorithm (denoted by Steep. Des.) is implemented directly in MATLAB for comparison purposes with successful algorithms, i.e., Quadratic Programming (denoted by Quad. P.), Interior-Point Method (denoted by IPM), Sequential Quadratic Programming (denoted by SQP), and the Active Set Method (denoted by Act. Set). More-sophisticated Riemannian algorithms, e.g., the Conjugate Gradient (denoted by Conj. Grad.) and Trust-Region (denoted by Trust R.), are implemented using the MATLAB manifold optimization toolbox “ManOpt" [37]. The performance of all algorithms is averaged over 1000 iterations and the mean value is presented herein.

The section is divided into three subsections: the first compares the running time of the proposed Riemannian methods against popular convex and non-convex algorithms. The second subsection solves a convex clustering problem [39] and

Table 3.2: Execution time of the doubly stochastic multinomial manifold.

n	20	40	60	80	100
Steep. Des.	0.099	0.134	0.148	0.197	0.216
Conj. Grad.	0.053	0.081	0.109	0.163	0.171
Trust R.	0.154	0.374	0.609	0.917	1.644
Quad. P.	0.821	101.4	1031	5745	9280
IPM	0.562	8.745	48.66	170.04	406.6
SQP	0.040	1.439	16.07	105.20	429.0
Act. Set	0.108	2.477	21.64	125.02	471.2

Table 3.3: Execution time of the symmetric multinomial manifold.

n	20	40	60	80	100
Steep. Des.	0.075	0.093	0.138	0.171	0.264
Conj. Grad.	0.045	0.049	0.062	0.087	0.094
Trust R.	0.069	0.120	0.162	0.357	0.786
Quad. P.	0.838	100.0	1004	5619	9076
IPM	0.620	9.641	53.20	191.0	456.7
SQP	0.034	0.510	3.531	16.42	66.95
Act. Set	0.089	1.145	7.043	26.09	98.33

testifies to the efficiency of the proposed algorithms against popular methods from the literature. Finally, the last subsection shows that the proposed framework outperforms a specialized algorithm [53] in finding the solution of a non-convex clustering problem.

Performance of the Proposed Manifolds

This section solves the following convex optimization problem $\min_{\mathbf{X}\in\mathcal{M}} ||\mathbf{A} - \mathbf{X}||_F^2$, wherein the manifold $\mathcal{M}$ is the doubly stochastic, symmetric, and definite multinomial manifold, respectively. For each of the experiment, matrix $\mathbf{A}$ is generated by $\mathbf{A} = \mathbf{M} + \mathbf{N}$ with $\mathbf{M} \in \mathcal{M}$ belonging to the manifold of interest and $\mathbf{N}$ is a zero-mean white Gaussian noise of unit variance. The variance of the noise has a negligible impact on the performance of the different algorithms as the optimization problem is convex.

The optimization problem is first solved using the steepest-descent algorithm to obtain the optimal solution $\mathbf{X}^\star$ with a predefined precision $\epsilon = 10^{-4}$. All other algorithms are executed using the same initialization and carried on until the desired optimal solution $\mathbf{X}^\star$ is reached with the same precision ϵ. The total execution time is displayed in the corresponding table.

Table 3.4: Execution time of the definite multinomial manifold.

n	20	40	60	80	100
Steep. Des.	0.067	0.092	0.131	0.174	0.279
Conj. Grad.	0.040	0.050	0.063	0.083	0.093
Trust R.	0.060	0.108	0.208	0.354	0.494
Quad. P.	0.799	99.79	1001	5631	9606
IPM	0.556	10.07	53.52	195.6	239.6
SQP	0.030	0.464	3.397	11.86	15.07
Act. Set	0.073	1.047	6.951	20.94	22.83

Table 3.2 illustrates the execution time of the proposed method in denoising a doubly stochastic matrix against the problem dimension. The table reveals a significant gain in the execution time for the steepest-descent method, conjugate gradient, and trust-region algorithms against all simulated algorithms. The gain in performance can be explained by the fact that the proposed method uses the geometry of the problem efficiently unlike generic optimization algorithms which convert the problem in a standard form and solve it using standard methods. The second-order method performs poorly as compared with the first-order method due to the fact that the expression of the Riemannian Hessian is complex to compute. The problem can be circumvented with the use of quasi-Newton methods, e.g., BHHH, BFGS.

Table 3.3 shows the running time of the proposed algorithms on the symmetric multinomial manifold against the problem size. One can note that the gain is more important than in Table 3.2. Indeed, the symmetric manifold enjoys a larger dimension reduction as compared with the doubly stochastic which makes the required ingredients easier to compute. One can note that the computation of the Riemannian Hessian on the symmetric multinomial manifold is more efficient than on the doubly stochastic manifold which is reflected in a better performance against state-of-the-art algorithms.

Table 3.4 displays the execution time for the definite multinomial manifold. The proposed Riemannian algorithms efficiently find the solution to the problem thanks to the fact that the optimal solution does not include vanishing entries or eigenvalues as pointed out previously. Given that such condition is not fulfilled, the performance of the positive-definite manifold is omitted, and a relaxed version using the symmetric manifold and regularization is presented.

Table 3.5: Performance of the Riemannian methods for convex clustering.

Algorithm	Run. Time	Var. of Inf.	Error Rate
Conj. Grad. $\mathcal{DP}_n$	13.8574	0.503	6.1%
Conj. Grad. $\mathcal{SP}_n$	9.148	0.540	6.3%
Trust R. $\mathcal{SP}_n$	128.0	0.503	6.1%
SQP	4363	0.525	6.3%
Act. Set	6409	0.576	6.3%

Similarity Clustering via Convex Programming

This section suggests using the proposed framework to solve a convex clustering problem [39]. Given an entry-wise non-negative similarity matrix $\mathbf{A}$ between n data points, the goal is to cluster these data points into r clusters. The similarity matrix for $n = 473$ images dogs from the Stanford Dogs Dataset [88] is obtained from crowdsourcing on Amazon Mechanical Turk in [89]. These images are divided into $r = 3$ different breeds, i.e., Norfolk Terrier (172), Toy Poodle (151) and Bouvier des Flandres (150). Around 15% (17260 edges) of the total number of entries in the matrix $\mathbf{A}$ are queried with an error rate of 22% as compared with the ground truth. Under the above conditions, the reference guarantees the recovery of the clusters by solving the following optimization problem:

$$\min_{\substack{\mathbf{X}\in\mathcal{SP}_n \\ \mathbf{X}\geq\mathbf{0}}} ||\mathbf{A}-\mathbf{X}||_F^2 + \lambda \text{Tr}(\mathbf{X}), \tag{3.19}$$

wherein λ is a regularizer parameter whose expression is derived in [39]. The optimal solution to the above problem is a block matrix (up to a permutation) of rank equal to the number of clusters. Due to such rank deficiency of the optimal solution, the definite positive manifold cannot be used to solve the above problem. Therefore, we reformulate the problem on $\mathcal{SP}_n$ by adding the adequate regularizers as below:

$$\min_{\mathbf{X}\in\mathcal{SP}_n} ||\mathbf{A}-\mathbf{X}||_F^2 + \lambda \text{Tr}(\mathbf{X}) + \rho(||\mathbf{X}||_* - \text{Tr}(\mathbf{X})), \tag{3.20}$$

wherein ρ is the regularization parameter. The expression of such regularizer can be obtained by expressing the Lagrangian of the original problem and deriving the expression of the Lagrange multipliers. However, this falls outside the scope of this paper. Clearly, the expression $||\mathbf{X}||_* - \text{Tr}(\mathbf{X}) = \sum_{i=1}^n |\lambda_i| - \lambda_i$ is positive and equal to zero if and only if all the eigenvalues are positive which concludes that $\mathbf{X}$ is positive. Similarly, the problem can be reformulated on $\mathcal{DP}_n$ as follows:

$$\min_{\mathbf{X}\in\mathcal{SP}_n} \mathsf{f}(\mathbf{X}) + \rho(||\mathbf{X}||_* - \text{Tr}(\mathbf{X})) + \mu(||\mathbf{X}-\mathbf{X}^{\text{T}}||_F^2), \tag{3.21}$$

where $\mathsf{f}(\mathbf{X})$ is the original objective function in (3.19) regularized with ρ and μ to promote positiveness and symmetry.

The nuclear norm of a matrix is a non-smooth function. However, the function is differentiable for full-rank matrices. Indeed, given a full-rank matrix $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, the gradient is computed in [90] as $\dfrac{\delta||\mathbf{X}||_*}{\delta\mathbf{X}} = \mathbf{U}\mathbf{V}^T$ which can be written as $\dfrac{\delta||\mathbf{X}||_*}{\delta\mathbf{X}} = \mathbf{U}\mathbf{V}^T = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-\frac{1}{2}}$. The non-smoothness arises for rank deficient matrices for which the gradient can be arbitrarily perturbed to produce an approximate gradient. More concretely, for singular matrices, the positive semi-definite matrix $\mathbf{X}^T\mathbf{X}$ is offset by $\epsilon\mathbf{I}$ to produce a positive-definite matrix. Therefore, we obtain the approximate gradient $\dfrac{\delta||\mathbf{X}||_*}{\delta\mathbf{X}} = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \epsilon\mathbf{I})^{-\frac{1}{2}}$ for some small perturbation $\epsilon > 0$. The above described approximate gradient is not valid for all applications. In general, one needs to perform a change a variable $||\mathbf{X}||_* \leq y \Leftrightarrow \begin{pmatrix}\mathbf{W}_1 & \mathbf{X} \\ \mathbf{X}^T & \mathbf{W}_2\end{pmatrix} \succeq \mathbf{0}$ and $\mathrm{Tr}(\mathbf{W}_1 + \mathbf{W}_2) \leq 2y$. However, the convex and smooth nature of our simulation allows convergence to the global optimum with the aforementioned approximate gradient.

Table 3.5 shows the running time for solving the convex clustering problem (3.21) along with the achieved variation of information and error rate for the different methods. Clearly, the proposed Riemannian optimization algorithms largely outperform the standard approach with gains ranging from 15 to 700 fold for the first-order methods. As the problem is convex, all the above methods are expected to produce similar results up to some numerically insignificant differences. Indeed, the precision of the proposed algorithms is satisfactory as they all achieve the same accuracy as state-of-the-art methods. Also, note that using the symmetric multinomial manifold produces better results. This can be explained by the fact that not only the objective function (3.20) is simpler than (3.21) but also by the fact that the manifold contains fewer degrees of freedom which makes the projections more efficient.

Clustering by Non-Convex Programming

This last part tests the performance of the proposed method for clustering by low-rank doubly stochastic matrix decomposition in the setting proposed in [53]. Given a similarity matrix as in the previous section, the authors in the above reference claim that a suitable objective function to determine the clusters structure is the

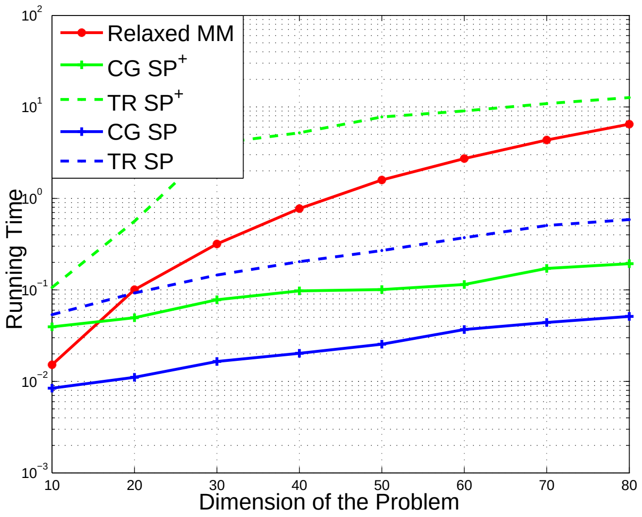


Figure 3.1: Running time of the different non-convex clustering algorithms.

following non-convex cost:

$$\max_{\substack{\mathbf{X}\in\mathcal{SP}_n \\ \mathbf{X}\geq\mathbf{0}}} \sum_{i,j} \mathbf{A}_{ij} \log\left(\sum_k \frac{\mathbf{X}_{ik}\mathbf{X}_{jk}}{\sum_v \mathbf{X}_{vk}}\right) + (\alpha - 1)\sum_{ij} \log\left(\mathbf{X}_{ij}\right).$$

The authors propose a specialized algorithm, known as “Relaxed MM", to efficiently solve the problem above. This section suggests solving the above problem using the positive and the symmetric multinomial manifold (with the proper regularization as shown in the previous subsection). In order to reach the same solution, all algorithms are initialized with the same value. The objective function achieved by the algorithm of [53] is taken as a reference, and the other algorithms stop as soon as their cost drops below such value.

Table 3.1 illustrates the running time of the different algorithms in order to reach the same solution. The plot reveals that the proposed framework is highly efficient in high dimension with significant gain over the specialized algorithm. The performance of the first-order method is noticeably better than the second-order one. This can be explained by the complexity of deriving the Riemannian Hessian. As stated before, in practical implementations, one would use an approximation of the Hessian to remedy to the complexity of computing the Hessian. Finally, one can note that the symmetric multinomial performs better than the positive one which can be explained by the fact that the optimal solution has vanishing eigenvalues which make the retraction on the cone of positive matrices non-efficient.

Chapter 4

LOW-RANK RIEMANNIAN METHODS FOR GRAPH-BASED CLUSTERING

[1] A. Douik and B. Hassibi. “Low-Rank Riemannian Optimization on Positive Semidefinite Stochastic Matrices with Applications to Graph Clustering”. In: *Proc. of the International Conference on Machine Learning (ICML’ 2018), Stockholm, Sweden*. Vol. 80. July 2018, pp. 1299–1308. URL: proceedings.mlr.press/v80/douik18a.html.

[2] A. Douik and B. Hassibi. “Non-Negative Matrix Factorization via Low-Rank Stochastic Manifold Optimization”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2019), Paris, France*. Vol. 1. 1. June 2019, pp. 497–501. DOI: 10.1109/ISIT.2019.8849441.

With the abundance of data, machine learning applications engaged increased attention in the last decade. An attractive approach to robustify the statistical analysis is to preprocess the data through clustering. While the previous chapter solves the problem using Riemannian optimization over the set of doubly stochastic matrices, this part develops a low-complexity Riemannian optimization framework for solving optimization problems on the set of positive semidefinite stochastic matrices. The low-complexity feature of the proposed algorithms stems from the factorization of the optimization variable $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ and deriving conditions on the number of columns of $\mathbf{Y}$ under which the factorization yields a satisfactory solution. The embedded and quotient geometries of the resulting Riemannian manifolds are investigated. In particular, the chapter explicitly derives the tangent space, Riemannian gradients and Hessians, and a retraction operator allowing the design of efficient first and second-order optimization methods for the graph-based clustering applications of interest. The numerical results reveal that the resulting algorithms present a clear complexity advantage as compared with state-of-the-art Euclidean and Riemannian approaches for graph clustering applications. Finally, the chapter extends the study to non-symmetric matrices by employing the factorization $\mathbf{X} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$. The resulting Riemannian manifolds and optimization algorithms allow solving a large class of clustering problems such as the non-nagative matrix factorization. The results on this section appear in the research papers [89], [91], and [92] and as such some of the text appears as it is in these publications.

4.1 Low-Rank Optimization on the Set of Stochastic Matrices

Overview, Related Work, and Contributions

As stated in the introduction, multiple non-convex optimization problems can be approximated using convex relaxations, e.g., non-negative matrix factorization [93, 94], compressive sensing [95], and matrix completion [34, 36]. Despite their convexity, solving these relaxed problems can often be a computation bottleneck in large-scale applications. Moreover, numerous relaxation techniques require expanding the dimension of the problem and the search space. For example, the celebrated lifting technique transforms an n-dimensional vector problem into a semidefinite $n \times n$ matrix problem. Luckily, these relaxations share an intrinsic property that greatly simplifies the computation, namely, the optimal solution is *low-rank*.

This chapter considers solving convex and non-convex optimization problems over the set of positive semidefinite stochastic matrices. Such problems arise in multiple communications [80, 77], signal processing [81, 76], and machine learning applications, notably in graph clustering and community detection [96, 39, 53, 54, 74, 55]. The factorization of the $n \times n$ optimization variable $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$, with $\mathbf{Y}$ being an $n \times p$ matrix, allows us to reduce the search space from its original n^2 dimension to np, which is extremely attractive for $p \ll n$. However, while convenient, the factorization turns the problem into a non-convex optimization with non-isolated solutions. Indeed, consider any solution $\mathbf{Y}$ and an orthogonal $p \times p$ matrix $\mathbf{O}$, the matrix $\mathbf{YO}$ also represents a solution, i.e., the solution is invariant by right multiplication by an orthogonal matrix.

Riemannian manifold optimization offers the required tools for designing specialized algorithms for solving the aforementioned problems with moderate complexity. In particular, Riemannian quotient manifolds allow grouping all solutions $\mathbf{YO}$ into the same equivalence class and solving the optimization problem over these classes. While this fact might be irrelevant for first-order methods, it can be crucial for the convergence of second-order algorithms, e.g., Newton's method [23]. Consider an optimization problem over the set of positive semidefinite stochastic matrices. This chapter suggests factorizing the optimization variable $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$, deriving conditions on p under which the factorization yields a satisfactory solution, and investigating the first and second-order geometries of the resulting manifolds so as to take advantage of Riemannian optimization to design highly efficient specialized algorithms.

Factorizing a low-rank matrix and casting the optimization problem into a non-convex form is well studied in the literature, particularly for solving semidefinite programs (SDPs). For example, the factorization $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ is suggested in [97] to solve the maximum cut problem. The authors in [98] exploit similar factorization to solve SDPs with fixed traces. More generally, [99] investigates low-rank factorization for solving SDPs in standard form. The authors study the equivalence between the original and reformulated problems and propose a method for choosing the rank of the factorization in such a way to reduce complexity while preserving the optimality of the solution. This thesis proposes similar theoretical guarantees for positive semidefinite stochastic matrices.

Taking advantage of both the low-rank factorization and the optimization over Riemannian manifolds, the authors in [100] propose a first-order Riemannian algorithm for solving optimization problems on the elliptope, i.e., positive semidefinite matrices with ones on the diagonal. The quotient manifold is investigated in [101] and an invariant metric that makes the manifold geodesically complete is derived. A simpler quotient structure is introduced in [102] to solve optimization problems with general trace constraints, including the elliptope and the spectahedron, by proposing a second-order algorithm with guaranteed quadratic convergence. This chapter follows a similar approach in the sense that a novel quotient structure is proposed to solve optimization problems in which the optimization variable is a positive semidefinite stochastic matrix.

The contribution of this chapter is two-fold, namely providing theoretical guarantees and designing optimization algorithms. The first part of the chapter exploits the factorization to cast the problem as a non-convex optimization with non-isolated solutions. The optimality conditions of the original problem and its non-convex reformulation are derived and analyzed. Such a theoretical study provides conditions under which the solution to the reformulated problem coincides with that of the original optimization problem. In particular, the analysis allows deriving a general procedure for solving the original problem in which the rank of the factorization is increased sequentially until the derived optimality conditions are satisfied.

The second part of the chapter focuses on proposing first and second-order algorithms for solving the reparametrized optimization problem using tools from the Riemannian optimization. In particular, the study of the manifolds geometries allows deriving first and second-order algorithms with guaranteed linear and quadratic convergence, respectively. Numerical experiments on clustering tasks reveal that the proposed algorithms present a clear complexity advantage over state-of-the-art

Euclidean and Riemannian approaches.

Finally, the analysis is extended to low-rank non-symmetric stochastic and doubly stochastic matrices so as to provide a unified framework for multiple clustering approaches, including the non-negative matrix factorization.

Problem Formulation

This section presents the optimization problems of interest and exploits the factorization to reparametrize them. After introducing the required notations and terminology, the optimality conditions of the original problem and its non-convex reformulation are derived and analyzed. The analysis reveals an efficient method for selecting the optimization rank while retaining equivalence with the original optimization problem.

Let $\mathbf{X} \in \mathbb{R}^{n \times n}$ be a real symmetric $n \times n$ matrix and define the objective function $g : \mathbb{R}^{n \times n} \longrightarrow \mathbb{R}$. While most applications possess a convex objective function, such assumption is not required herein. However, the thesis assumes that the objective function is smooth, i.e., at least differentiable for convex functions and at least twice differentiable for non-convex ones. This section in interested in solving the following problem

$$\min_{\mathbf{X} \in \mathbb{R}^{n \times n}} \mathsf{g}(\mathbf{X}) \tag{4.1a}$$

$$\text{s}.t.\ \mathbf{X}_{ij} \geq 0,\ 1 \leq i, j \leq n \tag{4.1b}$$

$$\sum_{j=1}^{n} \mathbf{X}_{ij} = 1,\ 1 \leq i \leq n, \tag{4.1c}$$

$$\mathbf{X} \succeq \mathbf{0}, \tag{4.1d}$$

wherein constraint (4.1b) underlines that the matrix is element-wise positive, constraint (4.1c) corresponds to the fact that the matrix is stochastic, and constraint (4.1d) insists that the matrix is positive semidefinite (PSD), i.e., the matrix is symmetric $\mathbf{X} = \mathbf{X}^{\mathrm{T}}$ and $\mathbf{V}^{\mathrm{T}}\mathbf{X}\mathbf{V} \geq 0$ for any vector $\mathbf{V} \in \mathbb{R}^{n}$.

The optimization problem (4.1) requires searching for a solution in a space of dimension $\frac{n(n+1)}{2} = \mathcal{O}(n^2)$ which can be intractable for large-scale problems, e.g., community detection with millions of individuals. Nevertheless, as stated earlier, the applications of interest in this paper share the intrinsic property that the optimal solution has a much smaller rank than the ambient dimension. Such structure can be used to reduce by and large the computation complexity of both the optimization problem and the involved matrix operations, e.g., eigenvalue decomposition, Cholesky factorization.

The rest of the paper assumes that the optimal solution $\mathbf{X}^*$ has a rank r. If such rank is known a priori, one may directly use the appropriate Riemannian geometry. However, in most applications, such rank is unknown in which case one needs to increase the size of the model p until the optimality conditions derived in the rest of this chapter are satisfied. The steps of the general approach for unknown rank r are described later.

The Non-Convex Reformulation and Optimality Conditions

As stated previously, solving problem (4.1) requires searching for a solution in an $\frac{n(n+1)}{2}$-dimensional space. To alleviate such computation burden, this section proposes using the low-rank decomposition $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ wherein $\mathbf{Y} \in \mathbb{R}^{n \times p}$. Hence, the reformulated optimization problem has the form

$$\min_{\mathbf{Y} \in \mathbb{R}^{n \times p}} \mathsf{g}(\mathbf{Y}\mathbf{Y}^{\mathrm{T}}) \tag{4.2a}$$

$$\text{s.}t. \; \mathbf{Y}\mathbf{Y}^{\mathrm{T}} \geq \mathbf{0} \tag{4.2b}$$

$$\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{1} = \mathbf{1}. \tag{4.2c}$$

Notice that the original PSD constraint (4.1d) is implicitly assumed in the structure $\mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ which is always a PSD matrix. The reparametrized problem requires searching for a solution in an $np \ll \frac{n(n+1)}{2}$ dimensional space. However, even under the assumption that the optimization problem (4.1) is convex, the reformulated problem (4.2) is non-convex.

Ideally, the decomposition rank p would coincide with the actual rank r of the solution $\mathbf{X}^*$. However, such rank being unknown in most applications, the rest of this section derives conditions under which the reformulated problem and the original one are equivalent. In other words, this section derives conditions under which an extreme point $\mathbf{Y}$ of (4.2) corresponds to an extreme point $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ of (4.1). To this end, first and second-order optimality conditions are derived. These results extend the findings of [99] to non-convex functions with inequality constraints.

The optimality condition of problem (4.1) is obtained from the first-order Karush-Kuhn-Tucker (KKT) conditions [5]. To simplify the analysis, the optimization problem is rewritten in a symmetric form as follows

$$\min_{\mathbf{X}\in\mathbb{R}^{n\times n}} \mathsf{g}(\mathbf{X}) \tag{4.3a}$$

$$\text{s.}t.\ \mathbf{X}+\mathbf{X}^{\mathrm{T}} \geq \mathbf{0} \tag{4.3b}$$

$$(\mathbf{X}+\mathbf{X}^{\mathrm{T}})\mathbf{1} = \mathbf{2} \tag{4.3c}$$

$$\mathbf{X} \geq \mathbf{0}. \tag{4.3d}$$

Let $\overline{\boldsymbol{\Psi}}$, σ and $\mathbf{S}_{\mathbf{X}}$ be the dual variables corresponding to constraints (4.3b), (4.3c), and (4.3d), respectively. The Lagrangian $\mathsf{L}_{\mathbf{X}}(\mathbf{X},\sigma,\mathbf{S}_{\mathbf{X}},\overline{\boldsymbol{\Psi}})$ can be written as

$$\begin{aligned}\mathsf{L}_{\mathbf{X}}(\mathbf{X},\sigma,\mathbf{S}_{\mathbf{X}},\overline{\boldsymbol{\Psi}}) &= \mathsf{g}(\mathbf{X}) + \langle\sigma, (\mathbf{X}+\mathbf{X}^{\mathrm{T}})\mathbf{1}-\mathbf{2}\rangle \\ &- \langle\overline{\boldsymbol{\Psi}}, \mathbf{X}+\mathbf{X}^{\mathrm{T}}\rangle - \langle\mathbf{S}_{\mathbf{X}},\mathbf{X}\rangle, \quad \overline{\boldsymbol{\Psi}} \geq \mathbf{0},\ \mathbf{S}_{\mathbf{X}} \geq \mathbf{0}.\end{aligned}$$

Differentiating the above Lagrangian with respect to the variable $\mathbf{X}$ gives

$$\mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{X})) + \sigma\mathbf{1}^{\mathrm{T}} + \mathbf{1}\sigma^{\mathrm{T}} - \mathbf{S}_{\mathbf{X}} - \overline{\boldsymbol{\Psi}} - \overline{\boldsymbol{\Psi}}^{\mathrm{T}} = 0$$

with the corresponding positiveness and complementary slackness constraints $\mathbf{S}_{\mathbf{X}} \geq \mathbf{0}$, $\overline{\boldsymbol{\Psi}} \geq \mathbf{0}$, $\mathbf{X}\odot\overline{\boldsymbol{\Psi}} = \mathbf{0}$, and $\mathbf{S}_{\mathbf{X}}\mathbf{X} = \mathbf{0}$. Let $\boldsymbol{\Psi} = \overline{\boldsymbol{\Psi}} + \overline{\boldsymbol{\Psi}}^{\mathrm{T}} \in \mathcal{S}^n$, the non-negativity constraints $\overline{\boldsymbol{\Psi}} \geq \mathbf{0}$ and $\boldsymbol{\Psi} \geq \mathbf{0}$ along with the complementary slackness conditions $\mathbf{X}\odot\overline{\boldsymbol{\Psi}} = \mathbf{0}$ and $\mathbf{X}\odot\boldsymbol{\Psi} = \mathbf{0}$ are equivalent. Finally, the uniqueness of the dual variables derives from the independence of the constraints [103].

Combining all the above, a solution $\mathbf{X} \in \mathbb{R}^{n\times n}$ is an extreme point of the optimization problem (4.1) if there exists the unique dual variables $\sigma \in \mathbb{R}^n$, $\mathbf{S}_{\mathbf{X}}, \boldsymbol{\Psi} \in \mathcal{S}^n$ such that the following equations hold

$$\mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{X})) + \sigma\mathbf{1}^{\mathrm{T}} + \mathbf{1}\sigma^{\mathrm{T}} = \mathbf{S}_{\mathbf{X}} + \boldsymbol{\Psi} \tag{4.4a}$$

$$\mathbf{X} \geq \mathbf{0} \tag{4.4b}$$

$$\mathbf{X} \geq \mathbf{0} \tag{4.4c}$$

$$\mathbf{X}\mathbf{1} = \mathbf{1} \tag{4.4d}$$

$$\mathbf{S}_{\mathbf{X}} \geq \mathbf{0} \tag{4.4e}$$

$$\boldsymbol{\Psi} \geq \mathbf{0} \tag{4.4f}$$

$$\mathbf{X}\odot\boldsymbol{\Psi} = \mathbf{0} \tag{4.4g}$$

$$\mathbf{S}_{\mathbf{X}}\mathbf{X} = \mathbf{0}, \tag{4.4h}$$

with constraint (4.4a) translating the fact $(\mathbf{X},\sigma,\mathbf{S}_{\mathbf{X}},\boldsymbol{\Psi})$ is a saddle point for the Lagrangian, (4.4b)-(4.4d) stating that the solution is a feasible point, (4.4e) and

(4.4f) representing the positiveness of the dual variables and (4.4g) and (4.4h) expressing the complementary slackness.

A similar analysis on the reparametrized optimization problem (4.2) allows to conclude that a solution $\mathbf{Y} \in \mathbb{R}^{n \times p}$ is an extreme point of (4.2) if there exists the unique dual variables $\lambda \in \mathbb{R}^n$ and $\mathbf{\Phi} \in \mathcal{S}^n$ satisfying the following

$$(\mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{Y}\mathbf{Y}^{\mathrm{T}})) + \lambda\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda^{\mathrm{T}} - \mathbf{\Phi})\mathbf{Y} = \mathbf{0} \tag{4.5a}$$

$$\mathbf{Y}\mathbf{Y}^{\mathrm{T}} \geq \mathbf{0} \tag{4.5b}$$

$$\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{1} = \mathbf{1} \tag{4.5c}$$

$$\mathbf{\Phi} \geq \mathbf{0} \tag{4.5d}$$

$$\mathbf{Y}\mathbf{Y}^{\mathrm{T}} \odot \mathbf{\Phi} = \mathbf{0}, \tag{4.5e}$$

with constraint (4.5a) translating the fact $(\mathbf{Y}, \lambda, \mathbf{\Phi})$ is a saddle point for the Lagrangian, (4.5b) and (4.5c) stating that the solution is a feasible point, (4.5d) representing the positiveness of the dual variable and (4.5e) expressing the complementary slackness.

Theorem 4.1 *An extreme point* $\mathbf{Y}$ *of the optimization problem (4.2) produces an extreme point* $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ *of the problem (4.1) if and only if the following holds*

$$\mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{Y}\mathbf{Y}^{\mathrm{T}})) + \lambda\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda^{\mathrm{T}} - \mathbf{\Phi} = \mathbf{S}_{\mathbf{Y}} \succeq \mathbf{0}, \tag{4.6}$$

wherein λ *and* $\mathbf{\Phi}$ *are the dual variable associated with* $\mathbf{Y}$.

Proof: Let $\mathbf{Y}$ be an extreme point of (4.2). Following the above analysis, there exists the dual variables σ and $\mathbf{\Phi}$ satisfying the conditions (4.5)-(4.5e). Let $\mathbf{S}_{\mathbf{Y}}$ be defined as $\mathbf{S}_{\mathbf{Y}} = \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{Y}\mathbf{Y}^{\mathrm{T}})) + \lambda\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda^{\mathrm{T}} - \mathbf{\Phi}$. The sufficient part of the proof is straightforward by considering the identification $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$, $\sigma = \lambda$, $\mathbf{S}_{\mathbf{X}} = \mathbf{S}_{\mathbf{Y}}$, and $\mathbf{\Psi} = \mathbf{\Phi}$. Indeed, one can clearly see that the optimality conditions (4.5)-(4.5e) with the positiveness of $\mathbf{S}_{\mathbf{Y}} \succeq \mathbf{0}$ result in the optimality conditions (4.4b)-(4.4h). Furthermore, by definition of $\mathbf{S}_{\mathbf{Y}}$, the constraint (4.4a) is satisfied which concludes by virtue of the optimality conditions in (4.4) that the point is an extreme point of the optimization problem (4.1).

Conversely, assume that $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ is an extreme point of (4.1) for some dual variables σ, $\mathbf{S}_{\mathbf{X}}$, and $\mathbf{\Psi}$. In order to show that $\mathbf{S}_{\mathbf{Y}} \succeq \mathbf{0}$, we exploit the uniqueness of the dual variables to establish the equality $\sigma = \lambda$ and $\mathbf{\Psi} = \mathbf{\Phi}$. From the optimality conditions in (4.4), by multiplying (4.4a) by $\mathbf{X1}$ and using the identity in (4.4d) and (4.4h), we obtain the following equation

$$(\sigma\mathbf{1}^{\mathrm{T}} + \mathbf{1}\sigma^{\mathrm{T}})\mathbf{1} = (\mathbf{\Psi} - \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})))\mathbf{1}.$$

Solving for σ in the above expression gives

$$\begin{aligned}(n\mathbf{I} + \mathbf{11}^{\mathrm{T}})\sigma &= (\mathbf{\Psi} - \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})))\mathbf{1}\\ \sigma &= \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{11}^{\mathrm{T}})(\mathbf{\Psi} - \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})))\mathbf{1}.\end{aligned}$$

Using the optimality conditions in (4.5) and combining the equations (4.5a) and (4.5c) yields the following equation for λ

$$\lambda = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{11}^{\mathrm{T}})(\mathbf{\Phi} - \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})))\mathbf{1}.$$

Exploiting the uniqueness of the variables, we conclude that $\sigma = \lambda$ and $\mathbf{\Phi} = \mathbf{\Psi}$. Finally, using the definition of $\mathbf{S_Y} = \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})) + \lambda\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda^{\mathrm{T}} - \mathbf{\Phi}$ with $\lambda = \sigma$ and $\mathbf{\Psi} = \mathbf{\Phi}$ gives $\mathbf{S_X} = \mathbf{S_Y}$. Therefore, the optimality condition (4.4e) results in $\mathbf{S_Y} \geq \mathbf{0}$ which concludes the proof. ■

The above first-order analysis and result in Theorem 4.1 are sufficient to declare optimality for optimization problems with a convex objective function g. For non-convex functions, the following theorem investigates the second-order optimality conditions.

Theorem 4.2 *A sufficient condition for an extreme point* $\mathbf{Y} \in \mathbb{R}^{n\times p}$ *of (4.2) to yield an extreme point* $\mathbf{X} = \mathbf{YY}^{\mathrm{T}}$ *of (4.1) is that* $\mathsf{Rank}\ (\mathbf{Y}) < p$.

Proof: The proof of this theorem relies on the second-order KKT conditions of the optimization problem (4.2). These conditions, combined with the rank deficiency assumption, allow showing that $\mathbf{S_Y}$ is positive which by virtue of Theorem 4.1 concludes that $\mathbf{X}$ is an extreme point of (4.1).

Proposition 4.1 *Let* $(\mathbf{Y}, \lambda, \mathbf{\Phi})$ *satisfying the first-order KKT conditions in (4.5) and let* $\mathbf{S_Y} = \mathsf{Grad}_{\mathbf{X}}(\mathsf{g}(\mathbf{YY}^{\mathrm{T}})) + \lambda\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda^{\mathrm{T}} - \mathbf{\Phi}$. *For all matrices* $\mathbf{Z} \in \mathbb{R}^{n\times p}$ *satisfying* $\mathbf{YZ}^{\mathrm{T}} = \mathbf{0}$*, it holds that* $\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{S_Y}\mathbf{Z}) \geq 0$.

Proof: Let the optimization problem (4.2) be rewritten in a more-systematic form as

$$\begin{aligned}\min_{\mathbf{Y}\in\mathbb{R}^{n\times p}} \ & \frac{1}{2}\mathsf{g}(\mathbf{Y}\mathbf{Y}^{\mathrm{T}})\\ s.t.\ & \mathsf{Tr}(\mathbf{E}_{ij}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}) \geq 0,\ 1 \leq i, j \leq n\\ & \mathsf{Tr}(\mathbf{E}_{i}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}}) = 1,\ 1 \leq i \leq n,\end{aligned}$$

wherein $\mathbf{E}_{ij}$ is the matrix with 1 on the (i, j)-th entry and zeros elsewhere and $\mathbf{E}_i$ is the matrix with ones on the i-th column and zeros elsewhere. The analysis of the second-order conditions of the above problem reveals the inequality

$$\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathsf{Hess}_{\ \mathbf{Y}}(\mathsf{Grad}_{\mathbf{Y}}\mathsf{L}_{\mathbf{Y}}(\mathbf{Y}, \lambda, \mathbf{\Phi})[\mathbf{Z}])) \geq 0, \tag{4.7}$$

for all matrix $\mathbf{Z} \in \mathbb{R}^{n\times p}$ satisfying the following

$$\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}(\mathbf{E}_{ij} + \mathbf{E}_{ji})\mathbf{Y}) = 0,\ 1 \leq i, j \leq n, \mathbf{\Phi}_{ij} > 0 \tag{4.8}$$

$$\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}(\mathbf{E}_{ij} + \mathbf{E}_{ji})\mathbf{Y}) = 0,\ 1 \leq i, j \leq n, \mathbf{\Phi}_{ij} = 0 \tag{4.9}$$

$$\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}(\mathbf{E}_{i} + \mathbf{E}_{i}^{\mathrm{T}})\mathbf{Y}) \geq 0,\ 1 \leq i \leq n. \tag{4.10}$$

In particular, note that the inequality in (4.7) is achieved for any matrix $\mathbf{Z} \in \mathbb{R}^{n\times p}$ satisfying $\mathbf{Y}\mathbf{Z}^{\mathrm{T}} = \mathbf{0}$. Indeed, the condition $\mathbf{Y}\mathbf{Z}^{\mathrm{T}} = \mathbf{0}$ implies the three conditions in (4.8), (4.9), and (4.10). Furthermore, noting that $\mathsf{Grad}_{\mathbf{Y}}\mathsf{L}_{\mathbf{Y}}(\mathbf{Y}, \lambda, \mathbf{\Phi}) = \mathbf{S}_{\mathbf{Y}}\mathbf{Y}$ and under the constraint $\mathbf{Y}\mathbf{Z}^{\mathrm{T}} = \mathbf{0}$, the left-hand side of (4.7) simplifies as follows

$$\begin{aligned}&\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathsf{Hess}_{\ \mathbf{Y}}(\mathsf{Grad}_{\mathbf{Y}}\mathsf{L}_{\mathbf{Y}}(\mathbf{Y}, \lambda, \mathbf{\Phi})[\mathbf{Z}]))\\ &\quad= \mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathsf{D}\ (\mathbf{S}_{\mathbf{Y}})[\mathbf{Z}]\mathbf{Y}) + \mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_{\mathbf{Y}}\mathbf{Z})\\ &\quad= \mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_{\mathbf{Y}}\mathbf{Z}),\end{aligned}$$

wherein the last equality is obtained from the fact that $\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathsf{D}\ (\mathbf{S}_{\mathbf{Y}})[\mathbf{Z}]\mathbf{Y}) = \mathsf{Tr}(\mathsf{D}\ (\mathbf{S}_{\mathbf{Y}})[\mathbf{Z}]\mathbf{Y}\mathbf{Z}^{\mathrm{T}}) = 0$. Therefore, we obtain $\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_{\mathbf{Y}}\mathbf{Z}) \geq 0$ for all $\mathbf{Z}$ such that $\mathbf{Y}\mathbf{Z}^{\mathrm{T}} = \mathbf{0}$ ■

The rest of the proof uses the result of Proposition 4.1 and shows that for a rank deficient solution $\mathbf{Y}$, the inequality $\mathsf{Tr}(\mathbf{Z}^{\mathrm{T}}\mathbf{S}_{\mathbf{Y}}\mathbf{Z}) \geq 0$ implies the positiveness of $\mathbf{S}_{\mathbf{Y}} \geq \mathbf{0}$ which by virtue of Theorem 4.1 results in $\mathbf{X}$ being an extreme point of (4.1). The above-mentionned steps are omitted as they mirrors the steps used in proving Theorem 7 in [102]. ■

4.2 The Embedded Low-Rank Positive Multinomial Manifold

Manifold, Tangent Space, and Metric

Recall that the embedded low-rank positive multinomial manifold $\mathcal{M}_p^n$ is defined as $\mathcal{M}_p^n = \{\mathbf{Y} \in \mathbb{R}_*^{n \times p} \mid \mathbf{Y}\mathbf{Y}^\mathrm{T} > \mathbf{0} \text{ and } \mathbf{Y}\mathbf{Y}^\mathrm{T}\mathbf{1} = \mathbf{1}\}$. The manifold $\mathcal{M}_p^n$ is seen as an embedded manifold of the set of non-singular matrices $\mathbb{R}_*^{n \times p}$. In other words, the manifold is regarded as an embedded structure in the non-compact Stiefel manifold $\mathbb{R}_*^{n \times p}$. Define the function $\mathsf{f}(\mathbf{Y}) = \mathsf{g}(\mathbf{Y}\mathbf{Y}^\mathrm{T})$ wherein the function g is defined in (4.2).

The tangent space $\mathcal{T}_\mathbf{Y}\mathcal{M}_p^n$ for a point $\mathbf{Y} \in \mathcal{M}_p^n$ is computed using a *two-way* inclusion. For an open interval I containing 0, let $\gamma : I \subseteq \mathbb{R} \to \mathcal{M}_p^n$ be a smooth curve in the embedded low-rank positive multinomial manifold satisfying $\gamma(0) = \mathbf{Y}$ and $\dot{\gamma}(0) = \frac{d\gamma(t)}{dt}\Big|_{t=0} = \xi_\mathbf{Y}$. The curve $\gamma(t)$ being in the manifold for $t \in I$, it satisfies $\gamma(t)\gamma(t)^\mathrm{T}\mathbf{1} = \mathbf{1}$. Taking derivative of the equation above, evaluating at $t = 0$, and substituting the values for $\gamma(0)$ and $\dot{\gamma}(0)$ yields the following characterization for the tangent space at $\mathbf{Y} \in \mathcal{M}_p^n$

$$\mathcal{T}_\mathbf{Y}\mathcal{M}_p^n \subseteq \{\xi_\mathbf{Y} \in \mathbb{R}^{n \times p} \mid (\xi_\mathbf{Y}\mathbf{Y}^\mathrm{T} + \mathbf{Y}\xi_\mathbf{Y}^\mathrm{T})\mathbf{1} = \mathbf{0}\} \tag{4.11}$$

The converse is obtained by extending the tangent space characterization of [23]. To that purpose, first introduce the rank and level set of a function. Let $F : \mathcal{M}_1 \to \mathcal{M}_2$ be a smooth function from a manifold $\mathcal{M}_1$ of dimension d_1 to the manifold $\mathcal{M}_2$ of dimension d_2. The rank of F at a point $\mathbf{X} \in \mathcal{M}_1$ is the dimension of the image of $\mathcal{T}_\mathbf{X}\mathcal{M}_1$ by the indefinite directional derivative $\mathsf{D}\left(F(\mathbf{X})\right)[.] : \mathcal{T}_\mathbf{X}\mathcal{M}_1 \to \mathcal{M}_2$, i.e., the rank is defined as the dimension of the manifold generated by $\mathsf{D}\left(F(\mathbf{X})\right)[\xi_\mathbf{X}]$ for all $\xi_\mathbf{X} \in \mathcal{T}_\mathbf{X}\mathcal{M}_1$. A function F is rank-constant if the rank is constant for all $\mathbf{X} \in \mathcal{M}_1$. Given $\mathbf{Y} \in \mathcal{M}_2$, the set $F^{-1}(\mathbf{Y})$ is called a level set or a fiber of F.

While the authors in [23] present a precise characterization of the tangent space of manifolds that can be expressed as a level set of some function, the following lemma extends the results for manifolds that contain the level set of some function.

Lemma 4.1 *Let $\mathcal{M}$ be a manifold embedded in the Euclidean space $\mathcal{E}$. Assuming that $\mathcal{M}$ contains the level set of a rank-constant function $F : \mathcal{E} \to \mathbb{R}^n$, the tangent space $\mathcal{T}_\mathbf{X}\mathcal{M}$ at $\mathbf{X} \in \mathcal{M}$ satisfies* $\mathsf{Ker}\left(\mathsf{D}\left(F(\mathbf{X})\right)\right) \subseteq \mathcal{T}_\mathbf{X}\mathcal{M}$.

Proof: The manifold $\mathcal{M}$ contains the level set of F. Let $\mathbf{Y}$ be the fiber such that $F^{-1}(\mathbf{Y}) \subseteq \mathcal{M}$ and define $\mathcal{M}'$ as $F^{-1}(\mathbf{Y}) = \mathcal{M}'$. Since $\mathcal{M}'$ is a level set of a smooth function, it represents a manifold. Note that $\mathcal{M}'$ satisfies the conditions of the theorem of [23] as it is a level set of a constant-rank function. Therefore, we obtain $\text{Ker}\ (\text{D}\ (F(\mathbf{X}))) = \mathcal{T}_{\mathbf{X}}\mathcal{M}'$.. Finally, by definition of $\mathcal{M}'$, we have $\mathcal{M}' \subseteq \mathcal{M}$ which implies that $\mathcal{T}_{\mathbf{X}}\mathcal{M}' \subseteq \mathcal{T}_{\mathbf{X}}\mathcal{M}$. Exploiting this last inequality yields the result $\text{Ker}\ (\text{D}\ (F(\mathbf{X}))) \subseteq \mathcal{T}_{\mathbf{X}}\mathcal{M}$. ■

Let $\mathbb{R}^{n\times p}_{*+}$ be the set of entry-wise and full-rank $n \times p$ matrices and define $F : \mathbb{R}^{n\times p}_{*+} \longrightarrow \mathbb{R}^n$ such that $F(\mathbf{Y}) = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{1}$. Given that $\mathbf{Y} \in \mathbb{R}^{n\times p}_{*+}$ with positive entries, it is clear that $\mathbf{Y}\mathbf{Y}^{\mathrm{T}} > \mathbf{0}$. Therefore, the manifold $\mathcal{M}^n_p$ includes the predecessor of $\mathbf{1}$ by F, i.e., $F^{-1}(\mathbf{1}) \subseteq \mathcal{M}^n_p$. The fact that F has a fixed rank can be readily obtained from the fact that $\mathbf{1}$ is a regular value of F, i.e., the rank of any $\mathbf{Y} \in F^{-1}(\mathbf{1})$ is equal to n. Using the result of Lemma 4.1, we conclude that

$$\begin{aligned} \text{Ker}\ (\text{D}\ (F(\mathbf{Y}))) \subseteq \mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p &\Leftrightarrow \\ \{\xi_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid \text{D}\ (F(\mathbf{Y}))[\xi_{\mathbf{Y}}] = \mathbf{0}\} \subseteq \mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p &\Leftrightarrow \\ \{\xi_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid (\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}})\mathbf{1} = \mathbf{0}\} \subseteq \mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p&. \end{aligned} \tag{4.12}$$

Combining (4.11) and (4.12) allows to conclude that the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p$ for a point $\mathbf{Y} \in \mathcal{M}^n_p$ is given by

$$\mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p = \{\xi_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid (\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}})\mathbf{1} = \mathbf{0}\}.$$

The dimension of the tangent space, and thus the dimension of the manifold, can readily be seen from its characterization. The tangent space is embedded in a np dimension Euclidean space with n independent linear equations which gives a dimension of $n(p-1)$.

Riemannian Gradient, Retraction, and Riemannian Hessian Expressions

The expression of the Riemannian gradient as a function of the Euclidean gradient is given in the following proposition

Proposition 4.2 *Let* $\text{Grad f}(\mathbf{Y})$ *be the Euclidean gradient of f at* $\mathbf{Y}$*, the Riemannian gradient* $\text{grad f}(\mathbf{Y})$ *is given by*

$$\text{grad f}(\mathbf{Y}) = \text{Grad f}(\mathbf{Y}) - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}, \tag{4.13}$$

with α being the n-dimensional vector obtained by

$$\alpha = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1} \times \\ \left(\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})^{\mathrm{T}}\right)\mathbf{1}.$$

Proof: Define $\Pi_{\mathbf{Y}}$ as the orthogonal projection from the ambient space $\mathbb{R}^{n\times p}$ to the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. As stated earlier, the Riemannian gradient at $\mathbf{Y}$ can be expressed as the orthogonal projection of the Euclidean gradient at $\mathbf{Y}$ onto the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ [23]. In order to derive an expression of the orthogonal projection $\Pi_{\mathbf{Y}}$, this proof first parametrizes the orthogonal complement of the tangent space. The orthogonal complement $\mathcal{T}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n$ of the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ in $\mathbb{R}^{n\times p}$ is parametrized by

$$\mathcal{T}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n = \{\eta \in \mathbb{R}^{n\times p} \mid \eta = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}\},$$

for some n-dimensional vector α. Indeed, let $\xi_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ and $\eta_{\mathbf{Y}} = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}$, we obtain

$$\begin{aligned} \langle \xi_{\mathbf{Y}}, \eta_{\mathbf{Y}} \rangle_{\mathbf{Y}} &= \mathsf{Tr}(\xi_{\mathbf{Y}}^{\mathrm{T}}(\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}) \\ &= \mathsf{Tr}(\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}}\mathbf{1}\alpha^{\mathrm{T}}) + \mathsf{Tr}(\mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}}\mathbf{1}\alpha^{\mathrm{T}}) \\ &\overset{(a)}{=} \mathsf{Tr}(\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}}\mathbf{1}\alpha^{\mathrm{T}}) - \mathsf{Tr}(\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}}\mathbf{1}\alpha^{\mathrm{T}}) = 0, \end{aligned} \tag{4.14}$$

wherein the equality in (a) is obtained from the fact that $\xi_{\mathbf{Y}}$ satisfies $(\xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}})\mathbf{1} = \mathbf{0}$. Therefore, we obtain that $\{\eta \in \mathbb{R}^{n\times p} \mid \eta = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}\} \subseteq \mathcal{T}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n$. Finally, note that $\mathsf{Dim}\ (\{\eta \in \mathbb{R}^{n\times p} \mid \eta = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}\}) = n$ which in addition to the dimension of tangent space of $n(p-1)$ concludes that $\mathcal{T}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n = \{\eta \in \mathbb{R}^{n\times p} \mid \eta = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}\}$.

We now derive an explicit expression for the orthogonal projection $\Pi_{\mathbf{Y}}$ onto the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. Let $\mathbf{Z} \in \mathbb{R}^{n\times p}$ be a point on the ambient space written as the sum of the component in the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ and its orthogonal complement as $\mathbf{Z} = \Pi_{\mathbf{Y}}(\mathbf{Z}) + \Pi_{\mathbf{Y}}^{\perp}(\mathbf{Z})$. Recall that a point $\Pi_{\mathbf{Y}}(\mathbf{Z})$ in the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ satisfies $(\Pi_{\mathbf{Y}}(\mathbf{Z})\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\Pi_{\mathbf{Y}}(\mathbf{Z})^{\mathrm{T}})\mathbf{1} = \mathbf{0}$. Furthermore, a point $\Pi_{\mathbf{Y}}^{\perp}(\mathbf{Z})$ on the orthogonal complement of the tangent space is parametrized by $\Pi_{\mathbf{Y}}^{\perp}(\mathbf{Z}) = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}$.

Therefore, we obtain

$$\begin{aligned}(\mathbf{Z}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{Z}^{\mathrm{T}})\mathbf{1} &= (\Pi^{\perp}_{\mathbf{Y}}(\mathbf{Z})\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\Pi^{\perp}_{\mathbf{Y}}(\mathbf{Z})^{\mathrm{T}})\mathbf{1} \\ &= \Big((\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}}(\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\Big)\mathbf{1} \\ &\overset{(a)}{=} n\alpha + \mathbf{1}\mathbf{1}^{\mathrm{T}}\alpha + \mathbf{Y}\mathbf{Y}^{\mathrm{T}}(n\alpha + \mathbf{1}\mathbf{1}^{\mathrm{T}}\alpha) \\ &\overset{(b)}{=} (n\mathbf{I} + 2\mathbf{1}\mathbf{1}^{\mathrm{T}} + n\mathbf{Y}\mathbf{Y}^{\mathrm{T}})\alpha, \end{aligned} \tag{4.15}$$

with both equalities in (a) and (b) being obtained from the fact that $\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{1} = \mathbf{1}$. Finally, notice the following identities

$$(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})(n\mathbf{I} + \mathbf{1}\mathbf{1}^{\mathrm{T}}) = n\mathbf{I} + 2\mathbf{1}\mathbf{1}^{\mathrm{T}} + n\mathbf{Y}\mathbf{Y}^{\mathrm{T}} \tag{4.16a}$$

$$(n\mathbf{I} + \mathbf{1}\mathbf{1}^{\mathrm{T}})^{-1} = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}}). \tag{4.16b}$$

Substituting (4.16a) and (4.16b) in (4.15) gives an expression for α as

$$\alpha = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1}(\mathbf{Z}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{Z}^{\mathrm{T}})\mathbf{1}.$$

Therefore, the orthogonal projection $\Pi_{\mathbf{Y}}$ from the ambient space $\mathbb{R}^{n\times p}$ onto the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p$ is given by

$$\Pi_{\mathbf{Y}}(\mathbf{Z}) = \mathbf{Z} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y},$$

with α defined by

$$\alpha = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1}(\mathbf{Z}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{Z}^{\mathrm{T}})\mathbf{1}.$$

As stated earlier, the final expression of the Riemannian gradient is obtained by projecting the Euclidean one onto the tangent space, i.e., $\mathsf{grad}\ \mathsf{f}(\mathbf{Y}) = \Pi_{\mathbf{Y}}(\mathsf{Grad}\ \mathsf{f}(\mathbf{Y}))$. Using the expression of the orthogonal projection, the Riemannian gradient can be expressed as

$$\mathsf{grad}\ \mathsf{f}(\mathbf{Y}) = \mathsf{Grad}\ \mathsf{f}(\mathbf{Y}) - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\mathbf{Y},$$

with α being the n-dimensional vector obtained by

$$\begin{aligned}\alpha = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1}\Bigg(&\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})\mathbf{Y}^{\mathrm{T}} \\ &+ \mathbf{Y}\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})^{\mathrm{T}}\Bigg)\mathbf{1}.\end{aligned}$$

■

As stated previously, the Riemannian Hessian $\mathsf{hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}]$ can be obtained by projecting the directional derivation $\mathsf{D}\ (\mathsf{grad}\ \mathsf{f}(\mathbf{Y}))\,[\xi_{\mathbf{Y}}]$ onto the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$, i.e., $\mathsf{hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}] = \Pi_{\mathbf{Y}}(\mathsf{D}\ (\mathsf{grad}\ \mathsf{f}(\mathbf{Y}))\,[\xi_{\mathbf{Y}}])$. Using the fact that $\mathsf{Hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}] = \mathsf{D}\ (\mathsf{Grad}\ \mathsf{f}(\mathbf{Y}))\,[\xi_{\mathbf{Y}}]$, the directional derivative of the Riemannian gradient is

$$\begin{aligned}\mathsf{D}\ (\mathsf{grad}\ \mathsf{f}(\mathbf{Y}))\,[\xi_{\mathbf{Y}}] = \mathsf{Hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}] &- (\dot{\alpha}\mathbf{1}^{\mathrm{T}} + \mathbf{1}\dot{\alpha}^{\mathrm{T}})\mathbf{Y} \\ &- (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\alpha^{\mathrm{T}})\xi_{\mathbf{Y}},\end{aligned} \tag{4.17}$$

with $\dot{\alpha}$ being defined by $\dot{\alpha} = \mathsf{D}\ (\alpha)\,[\xi_{\mathbf{Y}}]$. Define the constant matrix $\mathbf{A} = \frac{1}{n}(\mathbf{I} - \frac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})$ and recall that α is defined by

$$\alpha = \mathbf{A}(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1}\left(\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})^{\mathrm{T}}\right)\mathbf{1}.$$

The directional derivative of the quantity $(\mathbf{I} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}})^{-1} = \mathbf{B}$ can be obtained by exploiting the Kailath variant of Sherman-Morrison-Woodbury formula [104] which gives

$$\dot{\mathbf{B}} = \mathsf{D}\ (\mathbf{B})\,[\xi_{\mathbf{Y}}] = -\mathbf{B}(\mathbf{Y}\xi_{\mathbf{Y}}^{T} + \xi_{\mathbf{Y}}\mathbf{Y}^{T})\mathbf{B}.$$

Let the vector $\beta = \left(\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})^{\mathrm{T}}\right)\mathbf{1}$. Its directional derivative is given by

$$\begin{aligned}\dot{\beta} = \mathsf{D}\ (\beta)\,[\xi_{\mathbf{Y}}] = \Big(&\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})\xi_{\mathbf{Y}}^{\mathrm{T}} + \xi_{\mathbf{Y}}\mathsf{Grad}\ \mathsf{f}(\mathbf{Y})^{\mathrm{T}} \\ &+\mathsf{Hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}]\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathsf{Hess}\ \mathsf{f}(\mathbf{Y})\,[\xi_{\mathbf{Y}}]^{\mathrm{T}}\Big)\mathbf{1}.\end{aligned}$$

Finally, the expression of $\dot{\alpha}$ simplifies to $\dot{\alpha} = \mathbf{A}\mathbf{B}(\dot{\beta} - (\mathbf{Y}\xi_{\mathbf{Y}}^{T} + \xi_{\mathbf{Y}}\mathbf{Y}^{T})\mathbf{B}\beta)$ which allows to get the Riemannian Hessian by applying the projection operator $\Pi_{\mathbf{Y}}$ to the expression in (4.17).

Let $\mathsf{R}_{\mathbf{Y}}$ denote a retraction from the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ to the manifold $\mathcal{M}_p^n$. To derive an expression of such operator, recall that for an entry-wise positive matrix $\mathbf{A} \in \mathcal{S}^n$, there exists a unique diagonal matrix $\mathbf{D}$ with strictly positive entries such that $\mathbf{S} = \mathbf{DAD}$ is a doubly stochastic matrix. Such matrix is obtained by the DAD algorithm [86]. This result extends the Sinkhorn's theorem for symmetric matrices [84].

Let $\mathbb{R}^{n\times p}_{+/2} = \{\mathbf{Z} \in \mathbb{R}^{n\times p} \mid \mathbf{Z}\mathbf{Z}^{\mathrm{T}} > 0\}$ and introduce the projection $\Pi : \mathbb{R}^{n\times p}_{+/2} \to \mathcal{M}^n_p$ defined by $\Pi(\mathbf{Z}) = \mathbf{D}\mathbf{Z}$ wherein the diagonal matrix $\mathbf{D}$ is obtained from applying the DAD algorithm to the matrix $\mathbf{Z}\mathbf{Z}^{\mathrm{T}}$. This paper suggests the following retraction to project tangent vectors to the manifold

Theorem 4.3 *Let* $\mathsf{R}_{\mathbf{Y}} : \mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p \to \mathcal{M}^n_p$ *be defined by*

$$\mathsf{R}_{\mathbf{Y}}(\xi_{\mathbf{Y}}) = \Pi\Big(\mathbf{Y} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp\big\{(-\xi_{\mathbf{Y}})\big\}\Big), \tag{4.18}$$

with $\exp\big\{(\xi_{\mathbf{Y}})\big\}$ *begin the entry-wise exponential of the entries of the matrix* $\xi_{\mathbf{Y}}$. *The operator* $\mathsf{R}_{\mathbf{Y}}$ *is a well-defined retraction from the neighborhood* $\mathcal{N}_{\mathbf{0}}$ *of* $\mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p$ *to* $\mathcal{M}^n_p$.

Proof: In order to show that the operator suggested in the theorem represents a well-defined retraction, one needs to show that there exists an open interval $\mathcal{N}_{\mathbf{0}}$ around $\mathbf{0}$ such that for all tangent vectors $\xi_{\mathbf{Y}}$ in $\mathcal{N}_{\mathbf{0}} \subset \mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p$, for some neighborhood $\mathcal{N}_{\mathbf{0}}$, the following properties are satisfied

$$\mathsf{R}_{\mathbf{Y}}(\xi_{\mathbf{Y}}) \in \mathcal{M}^n_p \tag{4.19a}$$

$$\mathsf{R}_{\mathbf{Y}}(\mathbf{0}) = \mathbf{Y} \tag{4.19b}$$

$$\frac{d\mathsf{R}_{\mathbf{Y}}(\tau\xi_{\mathbf{Y}})}{d\tau}\bigg|_{\tau=0} = \xi_{\mathbf{Y}}. \tag{4.19c}$$

By definition of the retraction and the projection operator Π, it is sufficient to show that there exists a neighborhood $\mathcal{N}_{\mathbf{0}}$ such that the matrix $\mathbf{Y} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp\big\{(-\xi_{\mathbf{Y}})\big\}$ is in the set $\mathbb{R}^{n\times p}_{+/2}$ to conclude that $\mathsf{R}_{\mathbf{Y}}(\xi_{\mathbf{Y}})$ satisfies (4.19a). Let x_n be a real sequence converging to zero and define the matrix sequence $y_n = \mathbf{Y} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp\big\{(-x_n\xi_{\mathbf{Y}})\big\}$. Therefore, the matrix sequence y_n converges to $\mathbf{Y} \in \mathbb{R}^{n\times p}_{+/2}$. The set $\mathbb{R}^{n\times p}_{+/2}$ being both open and Hausdorff, there exists $N \in \mathbb{N}$ such that for all $n \geq N$, we have $y_n \in \mathbb{R}^{n\times p}_{+/2}$ which allows us to conclude that, for small-enough $\xi_{\mathbf{Y}}$, we have $\mathsf{R}_{\mathbf{Y}}(\xi_{\mathbf{Y}}) \in \mathcal{M}^n_p$.

The centering property (4.19b) can readily be seen from the definition of the retraction. Indeed, given that $e^0 = 1$, it is clear that $\mathsf{R}_{\mathbf{Y}}(\mathbf{0}) = \mathbf{Y}$. Finally, the local rigidity property (4.19c) is obtained from the study of the first-order perturbation of the projection operator Π. Note that the curve $\gamma(t) = \mathbf{Y} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp\big\{(-t\xi_{\mathbf{Y}})\big\}$ has a first-order approximation of $\gamma(t) = \mathbf{Y} + t\xi_{\mathbf{Y}} + o(t^2)$. Therefore, to obtain the derivative of $\mathsf{R}_{\mathbf{Y}}(\tau\xi_{\mathbf{Y}})$ around 0, it is sufficient to study the behavior of Π for small perturbations in the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. Recall that $\Pi(\mathbf{Z}) = \mathbf{DZ}$. Therefore, the first-order perturbation is given the diagonal matrix $\delta\mathbf{D}$ such that $\Pi(\mathbf{Z} + \delta\mathbf{Z}) = (\mathbf{D} + \delta\mathbf{D})(\mathbf{Z} + \delta\mathbf{Z})$. Since $\mathbf{Y} \in \mathcal{M}_p^n$, then $\Pi(\mathbf{Y}) = \mathbf{Y}$, i.e., $\mathbf{D} = \mathbf{I}$. Note that the first-order perturbation of $\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})$ can be obtained through the study of the first-order perturbation of the operator $\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})^{\mathrm{T}}$. Indeed, let $F : \mathcal{M}_1 \longrightarrow \mathcal{M}_2$ be a differential function with a first-order approximation $F(\mathbf{X} + \delta\mathbf{X}) = F(\mathbf{X}) + \mathsf{D}\left(F(\mathbf{X})\right)[\delta\mathbf{X}] + o(\delta\mathbf{X})^2$. Then, the operator $F(.)F^{\mathrm{T}}(.)$ has the following first-order expansion $F(\mathbf{X} + \delta\mathbf{X})F(\mathbf{X} + \delta\mathbf{X})^{\mathrm{T}} = F(\mathbf{X})F(\mathbf{X})^{\mathrm{T}} + F(\mathbf{X})\mathsf{D}\left(F(\mathbf{X})\right)[\delta\mathbf{X}]^{\mathrm{T}} + \mathsf{D}\left(F(\mathbf{X})\right)[\delta\mathbf{X}]F(\mathbf{X})^{\mathrm{T}} + o(\delta\mathbf{X})^2$ which allows to readily get the coefficients of F. The first-order perturbation of $\Pi(\mathbf{Y}+\xi_{\mathbf{Y}})\Pi(\mathbf{Y}+\xi_{\mathbf{Y}})^{\mathrm{T}}$ is given by

$$\begin{aligned}
&\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})^{\mathrm{T}} \\
&\quad = (\mathbf{I} + \delta\mathbf{D})(\mathbf{Y} + \xi_{\mathbf{Y}})(\mathbf{Y} + \xi_{\mathbf{Y}})^{\mathrm{T}}(\mathbf{I} + \delta\mathbf{D}) \\
&\quad \overset{(a)}{=} \mathbf{YY}^{\mathrm{T}} + \mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}} + \mathbf{YY}^{\mathrm{T}}\delta\mathbf{D} + \xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \delta\mathbf{D}\mathbf{YY}^{\mathrm{T}},
\end{aligned} \tag{4.20}$$

wherein the equality in (a) only keeps the first-order terms of the expansion. The following part of the proof uses the definition of the embedded manifold and its tangent space to conclude that $\delta\mathbf{D} = \mathbf{0}$. Recall that $\Pi(\mathbf{Z})\Pi(\mathbf{Z})^{\mathrm{T}}\mathbf{1} = \mathbf{1}$ and $(\mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}} + \xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}})\mathbf{1} = \mathbf{0}$ and define $\delta\mathbf{D}\mathbf{1} = \delta\mathbf{D}$. Multiplying (4.20) by $\mathbf{1}$ yields the following equation

$$\mathbf{1} = \mathbf{1} + \mathbf{0} + (\mathbf{YY}^{\mathrm{T}} + \mathbf{I})\delta\mathbf{D}.$$

Therefore, $\delta\mathbf{D}$ is in the null space of $(\mathbf{YY}^{\mathrm{T}} + \mathbf{I})$. However, since $\mathbf{YY}^{\mathrm{T}}$ is doubly stochastic, it is concluded that $\delta\mathbf{D} = \mathbf{0}$. Therefore, we obtain

$$\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})\Pi(\mathbf{Y} + \xi_{\mathbf{Y}})^{\mathrm{T}} = \mathbf{YY}^{\mathrm{T}} + \mathbf{Y}\xi_{\mathbf{Y}}^{\mathrm{T}} + \xi_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}},$$

which concludes that the first-order perturbation of Π is given by $\Pi(\mathbf{Y}+\xi_{\mathbf{Y}}) = \mathbf{Y}+\xi_{\mathbf{Y}}$. Therefore, we have $\mathsf{R}_{\mathbf{Y}}(\tau\xi_{\mathbf{Y}}) = \Pi(\gamma(\tau)) = \mathbf{Y} + \tau\xi_{\mathbf{Y}} + o(\mathbf{11}^{\mathrm{T}}\tau^2)$ and concludes that $\left.\dfrac{d\mathsf{R}_{\mathbf{Y}}(\tau\xi_{\mathbf{Y}})}{d\tau}\right|_{\tau=0} = \xi_{\mathbf{Y}}$. ■

4.3 The Quotient Low-Rank Positive Multinomial Manifold

The Quotient Structure of the Quotient Low-Rank Positive Multinomial

Due to the reformulation, the considered problem exhibits non-isolated solutions. Indeed, given a solution $\mathbf{Y} \in \mathcal{M}_p^n$ and an orthogonal matrix $\mathbf{O} \in \mathcal{O}^p$, the point $\mathbf{YO}$

represent another solution. Define the relationship $\sim$ on $\mathcal{M}_p^n$ such that

$$\mathbf{Y}_1 \sim \mathbf{Y}_2 \Leftrightarrow \exists\, \mathbf{O} \in \mathcal{O}^p \text{ s.t. } \mathbf{Y}_1\mathbf{O} = \mathbf{Y}_2.$$

Clearly, the relationship $\sim$ defines an equivalence relationship. Let the set $\overline{\mathcal{M}}_p^n = \mathcal{M}_p^n/\sim$, or equivalently $\overline{\mathcal{M}}_p^n = \mathcal{M}_p^n/\mathcal{O}^p$, be the quotient set. In order to show that the set $\overline{\mathcal{M}}_p^n = \mathcal{M}_p^n/\sim$ admits a manifold structure, it is sufficient to show that $\sim$ is regular [23], meaning that it satisfies the following three properties

1. $\mathsf{graph}\ (\sim)$ is an embedded submanifold of the product $\mathcal{M}_p^n \times \mathcal{M}_p^n$.
2. The projection $\pi_1 : \mathsf{graph}\ (\sim) \longrightarrow \mathcal{M}_p^n$ given by $\pi_1(\mathbf{Y}_1, \mathbf{Y}_2) = \mathbf{Y}_1$ is a submersion.
3. $\mathsf{graph}\ (\sim)$ is closed,

wherein $\mathsf{graph}\ (\sim) = \{(\mathbf{Y}_1, \mathbf{Y}_2) \in \mathcal{M}_p^n \times \mathcal{M}_p^n \mid \mathbf{Y}_1 \sim \mathbf{Y}_2\}$.

The proof that $\mathsf{graph}\ (\sim)$ is an embedded submanifold of the product $\mathcal{M}_p^n \times \mathcal{M}_p^n$ is omitted herein as it mirrors the steps in proving the same result for the Grassmann manifold. Indeed, note that $\mathsf{graph}\ (\sim) \subset \mathsf{graph}\ (\sim')$ with $\mathbf{Y}_1 \sim' \mathbf{Y}_2$ means that $\mathbf{Y}_1\mathbf{M} = \mathbf{Y}_2$ for some invertible $p \times p$ matrix $\mathbf{M}$. Therefore, $\mathsf{graph}\ (\sim)$ is an embedded submanifold as wanted.

For a matrix $\mathbf{Y} \in \mathcal{M}_p^n$, let $\mathbf{X} \in \mathbb{R}^{n \times p}$ satisfying $\mathbf{Y} + t\mathbf{X} \in \mathcal{M}_p^n$ for t in some interval $\mathcal{I}$ containing zero. The existence of $\mathbf{X}$ is guaranteed by the fact that $\mathcal{M}_p^n$ represents an open set, and thus it contains an open ball centered at each $\mathbf{Y}$. For an orthogonal matrix $\mathbf{O} \in \mathcal{O}^p$, define the curve $\gamma(t)$ by $\gamma(t) = (\mathbf{Y} + t\mathbf{X}, (\mathbf{Y} + t\mathbf{X})\mathbf{O})$. For $t \in \mathcal{I}$, it is clear that $\gamma(t) \in \mathsf{graph}\ (\sim)$. Furthermore, $\gamma(t)$ satisfies the property

$$\frac{d\pi_1(\gamma(t))}{dt}\bigg|_{t=0} = \frac{d(\mathbf{Y} + t\mathbf{X})}{dt}\bigg|_{t=0} = \mathbf{X}.$$

Therefore, π_1 is a submersion.

Finally, the closure of $\mathsf{graph}\ (\sim)$ can be obtained by noticing that $\mathsf{graph}\ (\sim)$ is the pre-image of $\mathbf{0}_{n\times n}$ by the function $(\mathbf{Y}_1, \mathbf{Y}_2) \mapsto \mathbf{Y}_1\mathbf{Y}_1^{\mathrm{T}} - \mathbf{Y}_2\mathbf{Y}_2^{\mathrm{T}}$. Indeed, note that $\mathbf{Y}_1\mathbf{Y}_1^{\mathrm{T}} = \mathbf{Y}_2\mathbf{Y}_2^{\mathrm{T}}$ implies that $\mathbf{Y}_1\mathbf{O} = \mathbf{Y}_2$ for some orthogonal matrix $\mathbf{O} \in \mathcal{O}^p$. Given that the function is continuous and that $\{\mathbf{0}_{n\times n}\}$ is a closed set, we conclude that $\mathsf{graph}\ (\sim)$ is closed.

Combining the three results above allows concluding that $\overline{\mathcal{M}}_p^n$ admits a unique manifold structure known as the quotient manifold of $\mathcal{M}_p^n$ by $\sim$. Points on $\overline{\mathcal{M}}_p^n$ are seen as equivalence classes denoted by $[\mathbf{Y}] = \overline{\mathbf{Y}}$ for $\mathbf{Y} \in \mathcal{M}_p^n$. Let $\pi : \mathcal{M}_p^n \to \overline{\mathcal{M}}_p^n$ be the canonical, or natural, projection of points to their equivalence class, i.e., $\pi(\mathbf{Y}) = \overline{\mathbf{Y}}$.

Note that $\mathsf{f}(\mathbf{Y}) = \mathsf{g}(\mathbf{Y}\mathbf{Y}^T)$ is invariant under $\sim$ as $\mathsf{f}(\mathbf{Y}_1) = \mathsf{f}(\mathbf{Y}_2)$ for all $\mathbf{Y}_1 \sim \mathbf{Y}_2$. Therefore, there exists a unique function $\overline{f} : \overline{\mathcal{M}}_p^n \to \mathbb{R}$, known as the projection of f, such that $\mathsf{f}(\mathbf{Y}) = \overline{f} \circ \pi(\mathbf{Y})$ for all $\mathbf{Y} \in \mathcal{M}_p^n$. The rest of this chapter is interested in studying the geometry of the quotient low-rank positive multinomial manifold in order to solve optimization problems of the form $\min_{\overline{\mathbf{Y}} \in \overline{\mathcal{M}}_p^n} \overline{f}(\overline{\mathbf{Y}})$.

The Riemannian Structure of the Quotient Low-Rank Positive Multinomial

Let $\overline{\mathbf{Y}} \in \overline{\mathcal{M}}_p^n$, the equivalence class $\pi^{-1}(\overline{\mathbf{Y}})$ can be represented by the following set $\pi^{-1}(\overline{\mathbf{Y}}) = \{\mathbf{Y}\mathbf{O} \mid \mathbf{O} \in \mathcal{O}^p\}$. Recall that the vertical space $\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n$ of $\overline{\mathbf{Y}}$ at $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$ is given by $\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n = \mathcal{T}_{\mathbf{Y}}\pi^{-1}(\overline{\mathbf{Y}})$. Let the function $\mathsf{F} : \mathbb{R}^{n\times p} \to \mathbb{R}^{n\times n}$ defined by $\mathsf{F}(\mathbf{Z}) = \mathbf{Y}\mathbf{Y}^T - \mathbf{Z}\mathbf{Z}^T$. Note that $\pi^{-1}(\mathbf{Y})$ is given by the level set of F at $\mathbf{0}_{n\times n}$. Indeed, each $\mathbf{Z}$ satisfying $\mathsf{F}(\mathbf{Z}) = \mathbf{0}$ implies that $\mathbf{Y}\mathbf{Y}^T = \mathbf{Z}\mathbf{Z}^T$, i.e., $\mathbf{Z} = \mathbf{Y}\mathbf{O}$ for some orthogonal matrix $\mathbf{O}$. Furthermore, it is straightforward to conclude that F is a constant-rank function from either the fact that $\mathbf{0}_{n\times n}$ is a regular value or by noting that F is a submersion onto the set of positive matrices. Therefore, the tangent space at $\mathbf{Y}$ is given by the kernel of the indefinite directional derivative (see Lemma 4.1). In other words, the vertical space is given by

$$\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n = \mathcal{T}_{\mathbf{Y}}\pi^{-1}(\overline{\mathbf{Y}}) = \{\xi_{\mathbf{Y}} \mid \xi_{\mathbf{Y}}\mathbf{Y}^T + \mathbf{Y}\xi_{\mathbf{Y}}^T = \mathbf{0}\}.$$

Recall that $\mathbf{Y} \in \mathbb{R}_*^{n\times p}$ is a full-rank matrix and define $\mathbf{Y}^\perp$ as any $n \times (n-p)$ matrix orthogonal complement of $\mathbf{Y}$ satisfying $\mathbf{Y}^T\mathbf{Y}^\perp = \mathbf{0}$, then any matrix $\xi_{\mathbf{Y}}$ can be written as $\xi_{\mathbf{Y}} = \mathbf{Y}\mathbf{M} + \mathbf{Y}^\perp\mathbf{K}$ for some $p \times p$ matrix $\mathbf{M}$ and some $(n-p) \times p$ matrix $\mathbf{K}$. Hence, the characterization $\xi_{\mathbf{Y}}\mathbf{Y}^T + \mathbf{Y}\xi_{\mathbf{Y}}^T = \mathbf{0}$ of $\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n$ can be rewritten as follows

$$\mathbf{Y}\mathbf{M}\mathbf{Y}^T + \mathbf{Y}^\perp\mathbf{K}\mathbf{Y}^T + \mathbf{Y}\mathbf{M}^T\mathbf{Y}^T + \mathbf{Y}\mathbf{K}^T(\mathbf{Y}^\perp)^T = \mathbf{0}.$$

Post and pre-multiplying the above equation by $\mathbf{Y}^T$ and $\mathbf{Y}$, respectively, gives $(\mathbf{Y}^T\mathbf{Y})\mathbf{M}(\mathbf{Y}^T\mathbf{Y}) = -(\mathbf{Y}^T\mathbf{Y})\mathbf{M}^T(\mathbf{Y}^T\mathbf{Y})$. Since $(\mathbf{Y}^T\mathbf{Y})$ is invertible, we obtain the following alternate representation of the vertical space of $\mathcal{M}_p^n$ at $\mathbf{Y}$

$$\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n = \{\mathbf{Y}\mathbf{M} \mid \mathbf{M} \in \mathcal{S}_{\mathsf{skew}}^p\}.$$

Theorem 4.4 *Consider the $\overline{\mathbf{Y}} \in \overline{\mathcal{M}}_p^n$. The bi-linear form defined on $\mathcal{T}_{\overline{\mathbf{Y}}}\overline{\mathcal{M}}_p^n \times \mathcal{T}_{\overline{\mathbf{Y}}}\overline{\mathcal{M}}_p^n$ by*

$$\langle \xi_{\overline{\mathbf{Y}}}, \eta_{\overline{\mathbf{Y}}} \rangle_{\overline{\mathbf{Y}}} = \mathsf{Tr}\left(\overline{\xi}_{\mathbf{Y}}^{\mathrm{T}} \overline{\eta}_{\mathbf{Y}}\right), \mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}}), \tag{4.21}$$

is a well-defined Riemannian metric that is compatible with $\sim$ and turns $\overline{\mathcal{M}}_p^n$ into a Riemannian quotient manifold. The horizontal distribution of $\overline{\mathbf{Y}} \in \overline{\mathcal{M}}_p^n$ at $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$ is

$$\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n = \{\eta_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n \mid \eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}\}.$$

Proof: It is sufficient to show that the Riemannian metric on the embedding space is compatible with the equivalence relationship $\sim$. Hence, this theorem is established by expressing the horizontal lifts $\overline{\eta}_{\mathbf{Y}}$ of tangent vectors $\eta_{\overline{\mathbf{Y}}}$ as function of an arbitrary representative $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$. Afterwards, it is shown that the inner product does not depend on the representative at which the horizontal lift is computed.

Recall that the horizontal distribution $\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n$ is given by the orthogonal complement of the vertical space $\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n$ in the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. The orthogonal complement of the vertical space is parameterized by the following

$$\begin{aligned}\mathcal{V}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n &= \{\eta_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid \langle \xi_{\mathbf{Y}}, \eta_{\mathbf{Y}} \rangle_{\mathbf{Y}} = 0,\ \forall\, \xi_{\mathbf{Y}} \in \mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n\} \\ &= \{\eta_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid \mathsf{Tr}\left(\eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y}\mathbf{M}\right) = 0,\ \forall\, \mathbf{M} \in \mathcal{S}_{\mathsf{skew}}^{p}\ \}.\end{aligned} \tag{4.22}$$

The set of skew-symmetric matrices can be re-parameterized as the difference of general matrices and its transpose, i.e., $\mathbf{M} = \mathbf{A} - \mathbf{A}^{\mathrm{T}}$ for some $\mathbf{A} \in \mathbb{R}^{p\times p}$. Therefore, the equality in (4.22) can be expressed as

$$\mathsf{Tr}\left(\eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y}\mathbf{A}\right) = \mathsf{Tr}\left(\mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}\mathbf{A}\right),\ \forall\, \mathbf{A} \in \mathbb{R}^{p\times p}.$$

Therefore, we obtain $\eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}$. Finally, the horizontal distribution is given by

$$\begin{aligned}\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n &= \mathcal{V}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n \cap \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n \\ &= \{\eta_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n \mid \eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}\}.\end{aligned}$$

To show that $\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n \oplus \mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n \subseteq \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$, let $\xi_{\mathbf{Y}} = \mathbf{Y}\mathbf{M} \in \mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n$, $\eta_{\mathbf{Y}} \in \mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n$, and define $\zeta_{\mathbf{Y}} = \xi_{\mathbf{Y}} + \eta_{\mathbf{Y}}$, then we have $(\zeta_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\zeta_{\mathbf{Y}}^{\mathrm{T}})\mathbf{1} = (\eta_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\eta_{\mathbf{Y}}^{\mathrm{T}})\mathbf{1} = \mathbf{0}$ which gives $\zeta_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. Finally, note that $\mathsf{Dim}\ (\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n) = \frac{p(p-1)}{2}$ and $\mathsf{Dim}\ (\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n) = np - \frac{p(p-1)}{2} - n$ which gives $\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n \oplus \mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n = \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$ as anticipated.

Let $\overline{\xi}_{\mathbf{Y}} \in \mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n$ represent the horizontal lift of $\xi_{\overline{\mathbf{Y}}} \in \mathcal{T}_{\overline{\mathbf{Y}}}\overline{\mathcal{M}_p^n}$ for $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$. The following lemma characterizes the horizontal lift within the same equivalence class with respect to the horizontal lift of an arbitrary representative

Lemma 4.2 *Let* $\overline{\mathbf{Y}} \in \overline{\mathcal{M}_p^n}$ *and* $\mathbf{Y} \in \mathcal{M}_p^n$ *such that* $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$. *For any* $\mathbf{O} \in \mathcal{O}^p$, *the following holds*

$$\overline{\xi}_{\mathbf{YO}} = \overline{\xi}_{\mathbf{Y}}\mathbf{O}.$$

Proof: Define $\mathcal{U}_{\mathbf{Y}} = \{[\mathbf{Z}] \mid \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{1} = c\mathbf{1}, \mathbf{Z}^{\mathrm{T}}\mathbf{Y} \in \mathcal{O}^p, c \in \mathbb{R}\}$ and let the function $\sigma_{\mathbf{Y}} : \mathcal{U}_{\mathbf{Y}} \to \mathbb{R}^{n\times p}$ be given by

$$\sigma_{\mathbf{Y}}([\mathbf{Z}]) = \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y}.$$

Note that $\sigma_{\mathbf{Y}}([\mathbf{Z}])$ can be written as $\sigma_{\mathbf{Y}}([\mathbf{Z}]) = \mathbf{ZO}$ for $\mathbf{O} = \mathbf{Z}^{\mathrm{T}}\mathbf{Y}$ being an orthogonal matrix. Therefore, $\sigma_{\mathbf{Y}}$ satisfies $\pi(\sigma_{\mathbf{Y}}([\mathbf{Z}])) = \pi(\mathbf{ZO}) = [\mathbf{Z}]$, i.e., it represents a right inverse of the natural projection π. Therefore, the operator $\mathsf{D}\,(\pi(\sigma_{\mathbf{Y}}([\mathbf{Z}]))) \circ \mathsf{D}\,(\sigma_{\mathbf{Y}}([\mathbf{Z}]))$ represents the identity map on $\mathcal{U}_{\mathbf{Y}}$. Furthermore, let $\mathbf{O} \in \mathcal{O}^p$ and note that the function satisfies the additional property $\sigma_{\mathbf{YO}}([\mathbf{Z}]) = \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{YO} = \sigma_{\mathbf{Y}}([\mathbf{Z}])\mathbf{O}$. Let $\mathcal{S}_{\mathbf{Y}}$ be the image of $\mathcal{U}_{\mathbf{Y}}$ by $\sigma_{\mathbf{Y}}$. The set $\mathcal{S}_{\mathbf{Y}}$ admits the following parameterization

$$\mathcal{S}_{\mathbf{Y}} = \{\mathbf{A} \in \mathbb{R}^{n\times p} \mid \mathbf{A} = \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y}, [\mathbf{Z}] \in \mathcal{U}_{\mathbf{Y}}\}. \tag{4.23}$$

The set $\mathcal{S}_{\mathbf{Y}}$ has the following explicit characterization as

$$\begin{aligned}\mathcal{S}_{\mathbf{Y}} = \{\mathbf{A} \in \mathbb{R}^{n\times p} \mid (\mathbf{A}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{A}^{\mathrm{T}})\mathbf{1} = c\mathbf{1}, c \in \mathbb{R},\\ \mathbf{A}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\mathbf{A}\}.\end{aligned} \tag{4.24}$$

Indeed, let $\mathbf{A} = \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y}$, then we have

$$\begin{aligned}(\mathbf{A}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{A}^{\mathrm{T}})\mathbf{1} &= (\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y}\mathbf{Y}^{\mathrm{T}} + \mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{Z}\mathbf{Z}^{\mathrm{T}})\mathbf{1}\\ &\overset{(a)}{=} \mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{1} + c\mathbf{1} = 2c\mathbf{1} = c'\mathbf{1},\end{aligned}$$

with the equality in (a) comes from the fact that $\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{1} = \mathbf{1}$. Furthermore, we have $\mathbf{A}^{\mathrm{T}}\mathbf{Y} = (\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y})^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\mathbf{Z}\mathbf{Z}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\mathbf{A}$. The equalities above conclude that $\mathcal{S}_{\mathbf{Y}}$ is included in the set described in (4.24). Finally, note that both representations in (4.23) and (4.24) have the same dimension $n(p-1) - \frac{p(p-1)}{2}$ which gives that (4.24) is a valid parameterization of $\mathcal{S}_{\mathbf{Y}}$. Such parameterization allows to get the indefinite directional derivative of $\sigma_{\mathbf{Y}}([\mathbf{Z}])$ as follows

$$\begin{aligned}\mathsf{D}\,\sigma_{\mathbf{Y}}([\mathbf{Z}]) &= \mathcal{T}_{\mathbf{A}}\mathcal{S}_{\mathbf{Y}}\\ &= \{\eta_{\mathbf{Y}} \in \mathbb{R}^{n\times p} \mid (\mathbf{Y}\eta_{\mathbf{Y}}^{\mathrm{T}} + \eta_{\mathbf{Y}}\mathbf{Y}^{\mathrm{T}})\mathbf{1} = \mathbf{0}, \eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}\}\\ &= \{\eta_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n \mid \eta_{\mathbf{Y}}^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\eta_{\mathbf{Y}}\} = \mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n,\end{aligned}$$

which concludes the characterization of the horizontal lift $\overline{\xi}_{\mathbf{Y}} = \mathsf{D}\,(\sigma_{\mathbf{Y}})([\mathbf{Z}])[\xi]$. Using the above equality, the horizontal lift over the equivalence class can be written as

$$\begin{aligned}\overline{\xi}_{\mathbf{YO}} &= \mathsf{D}\,(\sigma_{\mathbf{YO}})([\mathbf{Z}])[\xi] = \mathsf{D}\,(\sigma_{\mathbf{Y}}\mathbf{O})([\mathbf{Z}])[\xi]\\ &= \mathsf{D}\,(\sigma_{\mathbf{Y}})([\mathbf{Z}])[\xi]\mathbf{O} = \overline{\xi}_{\mathbf{Y}}\mathbf{O}.\end{aligned}$$

■

Using the result of Lemma 4.2, it becomes clear that the metric proposed in (4.21) is compatible with the equivalence relationship $\sim$. Indeed, for $\overline{\mathbf{Y}} \in \overline{\mathcal{M}_p^n}$ and any predecessor $\mathbf{Y}_1$ and $\mathbf{Y}_2 = \mathbf{Y}_1\mathbf{O}$ in $\pi^{-1}(\overline{\mathbf{Y}})$, we have the following

$$\begin{aligned}\langle \xi_{\overline{\mathbf{Y}}}, \eta_{\overline{\mathbf{Y}}}\rangle_{\overline{\mathbf{Y}}} &= \mathsf{Tr}\left(\overline{\xi}_{\mathbf{Y}_2}^{\mathrm{T}}\overline{\eta}_{\mathbf{Y}_2}\right)\\ &= \mathsf{Tr}\left(\overline{\xi}_{\mathbf{Y}_1\mathbf{O}}^{\mathrm{T}}\overline{\eta}_{\mathbf{Y}_1\mathbf{O}}\right)\\ &= \mathsf{Tr}\left(\mathbf{O}^T\overline{\xi}_{\mathbf{Y}_1}^{\mathrm{T}}\overline{\eta}_{\mathbf{Y}_1}\mathbf{O}\right)\\ &= \mathsf{Tr}\left(\overline{\xi}_{\mathbf{Y}_1}^{\mathrm{T}}\overline{\eta}_{\mathbf{Y}_1}\right),\ \forall\, \xi_{\overline{\mathbf{Y}}}, \eta_{\overline{\mathbf{Y}}} \in \mathcal{T}_{\overline{\mathbf{Y}}}\overline{\mathcal{M}_p^n}.\end{aligned}$$

Therefore, the Riemannian metric is independent of the chosen representation which concludes that $\overline{\mathcal{M}_p^n}$ admits a Riemannian manifold structure. ■

Geometry of the Quotient Low-Rank Positive Multinomial

Let $\Pi_{\mathbf{Y}}^{\mathcal{V}}$ and $\Pi_{\mathbf{Y}}^{\mathcal{H}}$ be the orthogonal projections from the ambient space to the vertical space $\mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n$ and horizontal space $\mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n$, respectively. Furthermore, let $\Pi_{\mathbf{Y}}$ be the orthogonal projection from the ambient space to the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$. Recall that the ambient space can be decomposed as $\mathbb{R}^{n\times p} = \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n \oplus \mathcal{T}_{\mathbf{Y}}^{\perp}\mathcal{M}_p^n$ wherein the tangent space can be expressed as $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n = \mathcal{V}_{\mathbf{Y}}\mathcal{M}_p^n \oplus \mathcal{H}_{\mathbf{Y}}\mathcal{M}_p^n$. Therefore, $\Pi_{\mathbf{Y}}^{\mathcal{H}}(\mathbf{Z})$ can be written as $\Pi_{\mathbf{Y}}^{\mathcal{H}}(\Pi_{\mathbf{Y}}(\mathbf{Z}))$ which reduces the study of $\Pi_{\mathbf{Y}}^{\mathcal{H}}$ to the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$.

Given a tangent vector $\xi_{\mathbf{Y}} \in \mathcal{T}_{\mathbf{Y}}\mathcal{M}_p^n$, it can be decomposed into a vertical and horizontal components as $\xi_{\mathbf{Y}} = \Pi_{\mathbf{Y}}^{\mathcal{V}}(\xi_{\mathbf{Y}}) + \Pi_{\mathbf{Y}}^{\mathcal{H}}(\xi_{\mathbf{Y}})$. Recall that $\Pi_{\mathbf{Y}}^{\mathcal{V}}(\xi_{\mathbf{Y}}) = \mathbf{YM}$ for some skew-symmetric matrix $\mathbf{M}$ and $\Pi^{\mathcal{H}}_{\mathbf{Y}}(\xi_{\mathbf{Y}})^{\mathrm{T}}\mathbf{Y} = \mathbf{Y}^{\mathrm{T}}\Pi^{\mathcal{H}}_{\mathbf{Y}}(\xi_{\mathbf{Y}})$. Therefore, matrix $\mathbf{M}$ satisfies the Sylvester equation $(\mathbf{Y}^{\mathrm{T}}\mathbf{Y})\mathbf{M}+\mathbf{M}(\mathbf{Y}^{\mathrm{T}}\mathbf{Y}) = \mathbf{Y}^{\mathrm{T}}\Pi_{\mathbf{Y}}(\mathbf{Z}) - \Pi_{\mathbf{Y}}(\mathbf{Z})^{\mathrm{T}}\mathbf{Y}$. Finally, rearranging the terms of the decomposition gives that the orthogonal projection of $\mathbf{Z}$ from the ambient space $\mathbb{R}^{n\times p}$ to the horizontal $\mathcal{H}_{\mathbf{Y}}\mathcal{M}^n_p$ is $\Pi^{\mathcal{H}}_{\mathbf{Y}}(\mathbf{Z}) = \Pi_{\mathbf{Y}}(\mathbf{Z}) - \mathbf{Y}\mathbf{M}$ where $\mathbf{M}$ is the solution to the above Sylvester equation. In other words, the quotient Riemannian gradient can be written as a function of the Euclidean and embedded Riemannian gradients as follows

$$\mathsf{grad}\,\overline{f}(\overline{\mathbf{Y}}) = \Pi^{\mathcal{H}}_{\mathbf{Y}}(\mathsf{Grad}\,\mathsf{f}(\mathbf{Y})) = \mathsf{grad}\,\mathsf{f}(\mathbf{Y}) - \mathbf{Y}\mathbf{M}, \tag{4.25}$$

with $\mathbf{M}$ being the solution to the Sylvester equation

$$(\mathbf{Y}^{\mathrm{T}}\mathbf{Y})\mathbf{M}+\mathbf{M}(\mathbf{Y}^{\mathrm{T}}\mathbf{Y}) = \mathbf{Y}^{\mathrm{T}}\mathsf{grad}\,\mathsf{f}(\mathbf{Y}) - \mathsf{grad}\,\mathsf{f}(\mathbf{Y})^{\mathrm{T}}\mathbf{Y}. \tag{4.26}$$

The expression of the Riemannian Hessian on the quotient space is obtained by projecting the directional derivative of the Riemannian gradient on the quotient manifold which gives

$$\mathsf{hess}\,\overline{f}(\overline{\mathbf{Y}})[\xi_{\overline{\mathbf{Y}}}] = \Pi^{\mathcal{H}}_{\mathbf{Y}}(\mathsf{hess}\,\mathsf{f}(\mathbf{Y})[\overline{\xi}_{\mathbf{Y}}] - \overline{\xi}_{\mathbf{Y}}\mathbf{M} - \mathbf{Y}\dot{\mathbf{M}}), \tag{4.27}$$

with $\mathbf{M}$ being the solution to (4.26) and $\dot{\mathbf{M}}$ satisfying

$$\begin{aligned}(\mathbf{Y}^{\mathrm{T}}\mathbf{Y})\dot{\mathbf{M}}+\dot{\mathbf{M}}(\mathbf{Y}^{\mathrm{T}}\mathbf{Y}) = {} & \overline{\xi}^{\mathrm{T}}_{\mathbf{Y}}\mathsf{grad}\,\mathsf{f}(\mathbf{Y}) - \mathsf{grad}\,\mathsf{f}(\mathbf{Y})^{\mathrm{T}}\overline{\xi}_{\mathbf{Y}} \\ & + \mathbf{Y}^{\mathrm{T}}\mathsf{hess}\,\mathsf{f}(\mathbf{Y})[\overline{\xi}_{\mathbf{Y}}] - \mathsf{hess}\,\mathsf{f}(\mathbf{Y})[\overline{\xi}_{\mathbf{Y}}]^{\mathrm{T}}\mathbf{Y} \\ & - 2(\overline{\xi}^{\mathrm{T}}_{\mathbf{Y}}\mathbf{Y}\mathbf{M} + \mathbf{M}\overline{\xi}^{\mathrm{T}}_{\mathbf{Y}}\mathbf{Y}).\end{aligned} \tag{4.28}$$

Let the retraction $\mathsf{R}_{\mathbf{Y}}$ be defined on the tangent space $\mathcal{T}_{\mathbf{Y}}\mathcal{M}^n_p$ by $\mathsf{R}_{\mathbf{Y}}(\xi_{\mathbf{Y}}) = \Pi(\mathbf{Y}+\xi_{\mathbf{Y}})$. The proof that the above operator represents a retraction on $\mathcal{M}^n_p$ is omitted as it mirrors the steps used in the proof of Theorem 4.3. Indeed, the proposed retraction represents the first-order approximation of the retraction in (4.18). Finally, consider the retraction

$$\overline{\mathsf{R}}_{\overline{\mathbf{Y}}}(\xi_{\overline{\mathbf{Y}}}) = \pi\Big(\Pi(\mathbf{Y}+\overline{\xi}_{\mathbf{Y}})\Big), \tag{4.29}$$

for $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$. The aforementioned operator represents a well-defined function as it does not depend on the representative $\mathbf{Y} \in \pi^{-1}(\overline{\mathbf{Y}})$. Indeed, let $\overline{\mathbf{Y}} \in \overline{\mathcal{M}^n_p}$ and consider a couple of representatives $\mathbf{Y}$ and $\mathbf{Y}\mathbf{O}$ in $\pi^{-1}(\overline{\mathbf{Y}})$. Let $\mathbf{D}$ be the diagonal matrix such that $\mathbf{D}(\mathbf{Y}+\overline{\xi}_{\mathbf{Y}})(\mathbf{Y}+\overline{\xi}_{\mathbf{Y}})^{\mathrm{T}}\mathbf{D}$ is doubly stochastic matrix, then

$$\begin{aligned}&\mathbf{D}(\mathbf{Y}\mathbf{O}+\overline{\xi}_{\mathbf{Y}\mathbf{O}})(\mathbf{Y}\mathbf{O}+\overline{\xi}_{\mathbf{Y}\mathbf{O}})^{\mathrm{T}}\mathbf{D} \\ &\quad = \mathbf{D}(\mathbf{Y}\mathbf{O}+\overline{\xi}_{\mathbf{Y}}\mathbf{O})(\mathbf{Y}\mathbf{O}+\overline{\xi}_{\mathbf{Y}}\mathbf{O})^{\mathrm{T}}\mathbf{D} \\ &\quad = \mathbf{D}(\mathbf{Y}+\overline{\xi}_{\mathbf{Y}})(\mathbf{Y}+\overline{\xi}_{\mathbf{Y}})^{\mathrm{T}}\mathbf{D}.\end{aligned}$$

In other words, $\pi\Big(\Pi(\mathbf{YO}+\bar{\xi}_{\mathbf{YO}})\Big) = \pi\Big(\Pi(\mathbf{Y}+\bar{\xi}_{\mathbf{Y}})\Big)$ which shows that $\overline{\mathsf{R}}_{\overline{\mathbf{Y}}}$ satisfies the local rigidity property. Therefore, we conclude that $\mathsf{R}_{\mathbf{Y}}$ is a well-defined retraction on the quotient manifold.

4.4 Algorithms and Performance of the Low-Rank Positive Multinomial Optimization Algorithms on the Positive Multinomial Manifold

This section exploits the previously studied embedded and quotient low-rank positive multinomial manifolds to propose a first and a second-order optimization algorithms on these manifolds. The optimality conditions are then utilized to solve the original optimization problem (4.1) by means of solving its non-convex reformulation (4.2).

Algorithm 4.1 Steepest-descent algorithm on the manifold $\mathcal{M}_p^n$.

```
Require: Manifold $\mathcal{M}_p^n$, function f, and tolerance $\epsilon$.
Initialize $p = p_0$.
Initialize $\mathbf{S}_{\mathbf{Y}} = -\mathbf{I}$.
while $\mathbf{S}_{\mathbf{Y}} \prec \mathbf{0}$ do
    Initialize $\mathbf{Y}$ randomly in $\mathcal{M}_p^n$.
    while $||\text{grad f}(\mathbf{Y})|| \geq \epsilon$ do
        Find Riemannian Gradient $\text{grad f}(\mathbf{Y})$ using (4.13).
        Set descent direction $\xi_{\mathbf{Y}} = -\frac{\text{grad f}(\mathbf{Y})}{||\text{grad f}(\mathbf{Y})||_{\mathbf{Y}}}$
        Find $\delta$ using the Armijo's backtracking procedure.
        Retract $\mathbf{Y} = \mathsf{R}_{\mathbf{Y}}(\delta\xi_{\mathbf{Y}})$ using (4.18).
    end while
    Solve the convex program in (4.30) to get $\lambda$ and $\mathbf{\Phi}$.
    Compute $\mathbf{S}_{\mathbf{Y}}$ using (4.6).
    Set $p = p + 1$.
end while
Output $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\text{T}}$.
```

This manuscript suggests solving a sequence of optimization problems with increasing dimension p, i.e., an increasing rank of the solution, until the theoretical guarantees in Theorem 4.1 are satisfied. Recall that the condition that $\mathbf{S}_{\mathbf{Y}}$ being positive semidefinite is sufficient to guarantee the optimality of $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\text{T}}$ for a convex objective function. Therefore, initializing p with some p_0 (usually $p_0 = 1$), the algorithm solve the optimization problem (4.2) using the Riemannian geometry tool. In particular, starting at a random point $\mathbf{Y}$ in the manifold, the tangent space is computed, and the Euclidean gradient is projected to obtain a descent direction $\xi_{\mathbf{Y}}$. For enhanced performance, the step size δ is optimized through the Armijo's backtracking procedure [23]. Afterwards, the intermediate point on the tangent space is retracted to the manifold. The process is repeated until the gradient vanishes. After obtaining the solution $\mathbf{Y}^*$, the dual variable $\mathbf{\Phi}$ is obtained by solving the convex problem

$$\begin{aligned} \min_{\mathbf{\Phi}\in\mathcal{S}^n} \; & ||(\mathsf{Grad}\ \mathsf{f}(\mathbf{Y}) + \lambda(\mathbf{\Phi})\mathbf{1}^{\mathrm{T}} + \mathbf{1}\lambda(\mathbf{\Phi})^{\mathrm{T}} - \mathbf{\Phi})\mathbf{Y}||_F^2 \\ \text{s.}t.\ & \mathbf{\Phi} \geq \mathbf{0} \\ & \mathbf{Y}\mathbf{Y}^{\mathrm{T}} \odot \mathbf{\Phi} = \mathbf{0}, \end{aligned} \tag{4.30}$$

with $\lambda(\mathbf{\Phi}) = \dfrac{1}{n}(\mathbf{I} - \dfrac{1}{2n}\mathbf{1}\mathbf{1}^{\mathrm{T}})(\mathbf{\Phi} - \mathsf{Grad}\ \mathsf{f}(\mathbf{Y}))\mathbf{1}$. Given the unique solution $\mathbf{\Phi}$, the second dual variable λ is obtained by the expression above which allows to compute matrix $\mathbf{S}_{\mathbf{Y}}$ and check its non-negativity. In case of success, the optimal $\mathbf{X}^* = \mathbf{Y}^*(\mathbf{Y}^*)^{\mathrm{T}}$ is returned. Otherwise, the process is repeated for an increased value of p. The steps of the algorithm are summarized in Algorithm 4.1.

Algorithm 4.2 Newton's method on the manifold $\overline{\mathcal{M}}_p^n$.

```
Require: Manifold $\overline{\mathcal{M}}_p^n$, function $\mathsf{f}$, and tolerance $\epsilon$.
Initialize $p = p_0$.
Initialize $\mathbf{S}_{\mathbf{Y}} = -\mathbf{I}$.
while $\mathbf{S}_{\mathbf{Y}} \prec \mathbf{0}$ do
    Initialize $\mathbf{Y}$ in $\mathcal{M}_p^n$ representative of $\pi(\mathbf{Y}) \in \overline{\mathcal{M}}_p^n$.
    while $||\mathsf{grad}\ \mathsf{f}(\mathbf{Y})|| \geq \epsilon$ do
        Find Riemannian Gradient $\mathsf{grad}\ \mathsf{f}(\mathbf{Y})$ and Hessian $\mathsf{hess}\ \mathsf{f}(\mathbf{Y})$ on $\mathcal{M}_p^n$ using (4.13) and (4.17), respectively.
        Find Riemannian Gradient $\mathsf{grad}\ \overline{f}(\mathbf{Y})$ and Hessian $\mathsf{hess}\ \overline{f}(\mathbf{Y})$ on $\overline{\mathcal{M}}_p^n$ using (4.25), (4.26), (4.27), and (4.28).
        Find descent direction $\overline{\xi}_{\mathbf{Y}}$ satisfying $\mathsf{hess}\ \mathsf{f}(\mathbf{Y})[\overline{\xi}_{\mathbf{Y}}] = \mathsf{grad}\ \mathsf{f}(\mathbf{Y})$
        Find $\delta$ using the Armijo's backtracking procedure.
        Retract $\mathbf{Y} = \mathsf{R}_{\mathbf{Y}}(\delta\overline{\xi}_{\mathbf{Y}})$ using (4.29).
    end while
    Solve the convex program in (4.30) to get $\lambda$ and $\mathbf{\Phi}$.
    Compute $\mathbf{S}_{\mathbf{Y}}$ using (4.6).
    Set $p = p + 1$.
end while
Output $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$.
```

Newton's method on the quotient manifold follows similar steps as the one in on embedded manifolds by properly substituting the geometrical objects. The algorithm sequentially increases the rank of the solution and solves the optimization problem using Newton's method on the manifold $\overline{\mathcal{M}}_p^n$. The Newton's step on the quotient

Table 4.1: Performance of the Riemannian methods for non-convex clustering.

Algorithm	Running Time	Var. of Inf.	Accuracy
CVX	3183.060 s	0.5404	6.3%
ALM	2.848651 s	0.8688	12.68%
CG on $\mathcal{M}^n$	6.121646 s	0.5543	6.7%
CG on $\mathcal{M}_p^n$	4.777171 s	0.5403	6.3%
CG on $\overline{\mathcal{M}}_p^n$ s	3.813541	0.5501	6.5%

manifold is accomplished by computing the Riemannian gradient and Hessian and finding the tangent vector $\xi_{\mathbf{Y}}$ satisfying $\mathsf{hess}\ \mathsf{f}(\mathbf{Y})[\overline{\xi}_{\mathbf{Y}}] = \mathsf{grad}\ \mathsf{f}(\mathbf{Y})$. Afterwards, the step size δ is computed, and the point retracted onto the manifold. In a similar manner as before, the algorithm stop for the first rank p for which the matrix $\mathbf{S}_{\mathbf{Y}}$ is non-negative. The steps are summarized in Algorithm 4.2.

Similarity Clustering via Convex Programming

This chapter uses the same real-world data obtained in the previous chapter through crowdsourcing on Amazon Mechanical Turk [105]. It compares the performance of the proposed Conjugate Gradient (CG) and the Trust-Region (TR) methods on both the embedded and the quotient manifold. The performance of the proposed algorithms is tested again the generic convex solver CVX [106] and the symmetric multinomial $\mathcal{M}^n = \{\mathbf{X} \in \mathcal{S}^n \mid \mathbf{X} > \mathbf{0}, \mathbf{X} > \mathbf{0}, \mathbf{X1} = \mathbf{1}\}$ introduced in the previous chapter. All simulations are carried out using the MATLAB toolbox Manopt [37] on an Intel Xeon Processor E5-1650 v4 (15M cache, 3.60 GHz) computer with 32Gb 2.4 GHz DDR4 RAM.

This part reveals the cluster structure by solving the following convex optimization problem whose theoretical guarantees are studied in [39]

$$\min_{\mathbf{X} \in \mathcal{M}} \frac{1}{2} ||\mathbf{A} - \mathbf{X}||_{\overline{F}}^2 + \lambda \mathsf{Tr}(\mathbf{X}),$$

wherein the restricted Frobenius $\overline{F}$ norm is computed only over the observed entries of $\mathbf{A}$. The optimization problem is solved using the numerical optimization toolbox CVX [106], a specialized approximate and fast algorithm [107], known as augmented Lagrange multipliers (ALM), and the symmetric multinomial. Afterwards, the same problem is solved by reformulating $\mathbf{X} = \mathbf{Y}\mathbf{Y}^{\mathrm{T}}$ and using our proposed methods on the embedded and the quotient manifold.

The quality of the recovery is attested through the computation of the variation of information [108] between the reached cluster structure and the ground truth.

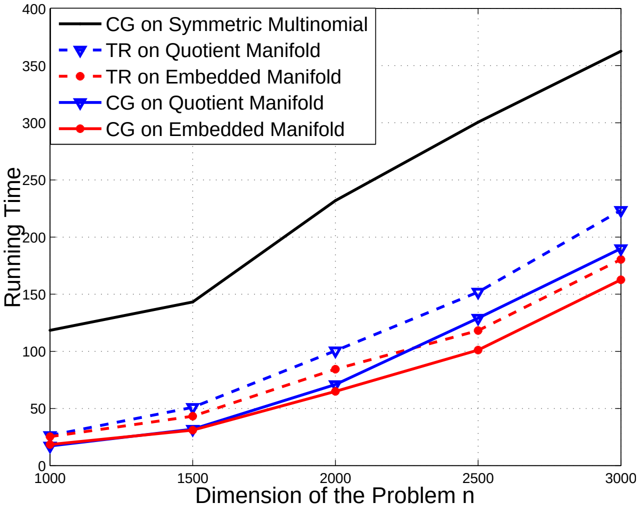


Figure 4.1: Accuracy of Riemannian methods for clustering a large system.

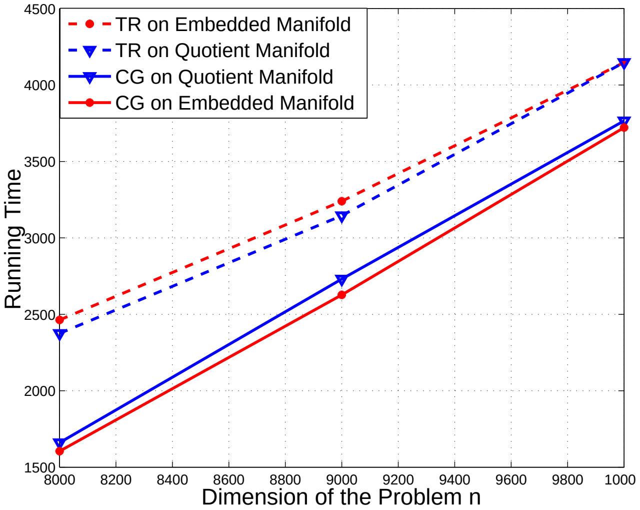


Figure 4.2: Running time of Riemannian methods for clustering a large system.

Table 4.1 illustrates the running time and the performance of the above-mentioned optimization methods. To be fair in comparison, only the conjugate gradient algorithm is employed over the different manifolds. Clearly, the proposed methods considerably outperform the generic solver CVX and the Riemannian symmetric multinomial in term of running time while maintaining the quality of the solution. The simulation also shows that the quotient manifold provides better results than its embedded counterpart which is expected as the quotient manifold reduces the dimension of the ambient space by grouping all equivalent solutions.

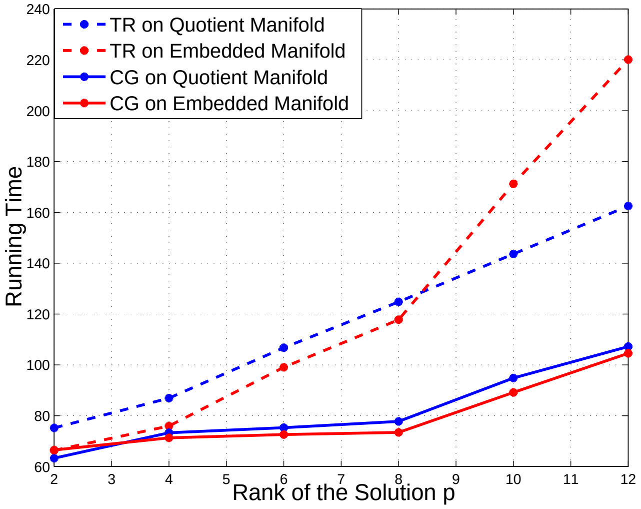


Figure 4.3: Accuracy of Riemannian methods against the number of clusters.

High Dimension Community Detection

This subsection proposes solving the clustering problem for a large number of entries n ,e.g., large number of dogs, using synthetic data. In particular, the crowdsourcing part is simulated by sampling from a stochastic block model similar to the one studied in [39] to obtain the similarity matrix. The number of clusters, e.g., the number of breeds of dogs, is also variable so as to study multiple scenarios. The size of clusters is chosen randomly from a set of predefined sizes for each dimension such that the recovery is theoretically guaranteed. Furthermore, the parameters of the stochastic block model are selected so that the theoretical guarantees proposed in [39] are valid which is further confirmed by an almost null total variation of information between the ground truth and the reached solution. The first part of these simulations compares the time performance of the proposed first and second-order methods on the embedded and quotient manifolds against the performance achieved by a first-order method on the symmetric multinomial $\mathcal{M}^n$. The second part shows the performance of the proposed solution against a system of huge dimension. For such large dimension, neither the generic CVX nor the specialized Riemannian symmetric multinomial are applicable. The final part plots the running time of the suggested methods against the number of clusters p for a fixed dimension n.

Figure 4.1 plots the running time of the proposed first and second-order methods again the dimension of the problem n for clusters scaling as $p = 4n/1000$. As a base of comparison, this subsection plots the performance of the conjugate gradient

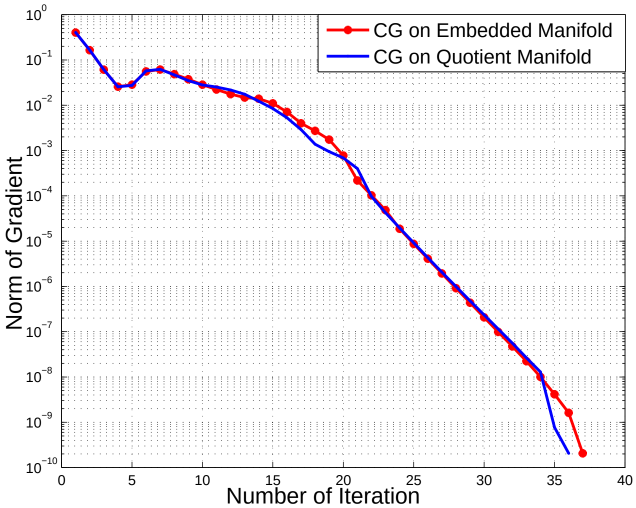


Figure 4.4: Convergence rate of the conjugate gradient algorithm on $\mathcal{M}_p^n$ and $\overline{\mathcal{M}}_p^n$.

algorithm on the symmetric multinomial. The running time of the second-order method on the symmetric multinomial requires solving an $n \times n$ system of linear equations which largely deteriorate its performance and thus is omitted in these simulations. Figure 4.1 clearly displays that the proposed methods achieve the same performance with drastically lower running time. The behavior is further illustrated in Figure 4.2 wherein the system dimension is very large $8000 \leq n \leq 10000$ for a number of clusters $p = n/1000$. The configuration of Figure 4.1 is prohibitively complex to run either CVX or the symmetric multinomial. Nevertheless, our proposed methods achieve the optimal solution in reasonable running time.

Figure 4.3 plots the performance of the proposed algorithms in clustering large data sets versus the number of clusters. As shown in the analysis in the manuscript, the dimension of the suggested manifold increases with the rank p, i.e., number of clusters. Such fact is attested by Figure 4.3. The figure especially displays that the second-order method is more sensitive to a change in p as an $p \times p$ linear systems needs to be solved at each iteration for these second-order methods.

Optimization with Unknown Rank p

This final set of simulation attests the performance of Algorithm 4.1 and Algorithm 4.2 in finding the optimal solution for an unknown rank p. A matrix $\mathbf{A}$ is generated from a low-rank p doubly stochastic matrix and corrupted with a zero-mean white Gaussian noise with unitary total power. This part solves the following denoising problem $\min_{\mathbf{Y} \in \mathcal{M}_p^n} ||\mathbf{A} - \mathbf{Y}\mathbf{Y}^{\mathrm{T}}||_F^2$. Assuming the solution rank p is known,

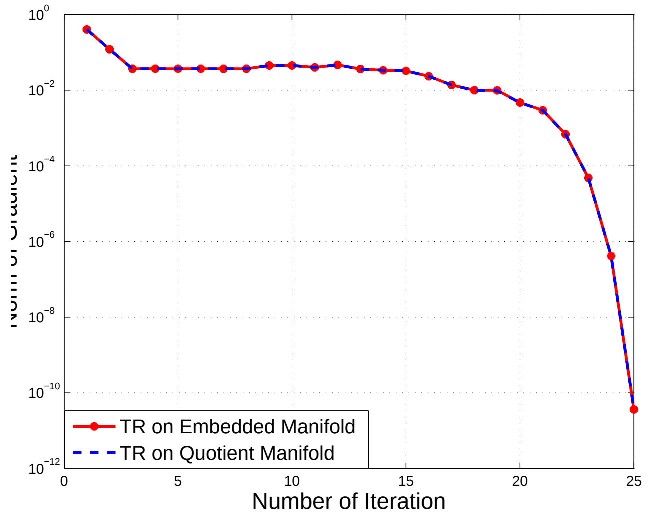


Figure 4.5: Convergence rate of the trust-region algorithm on $\mathcal{M}_p^n$ and $\overline{\mathcal{M}}_p^n$.

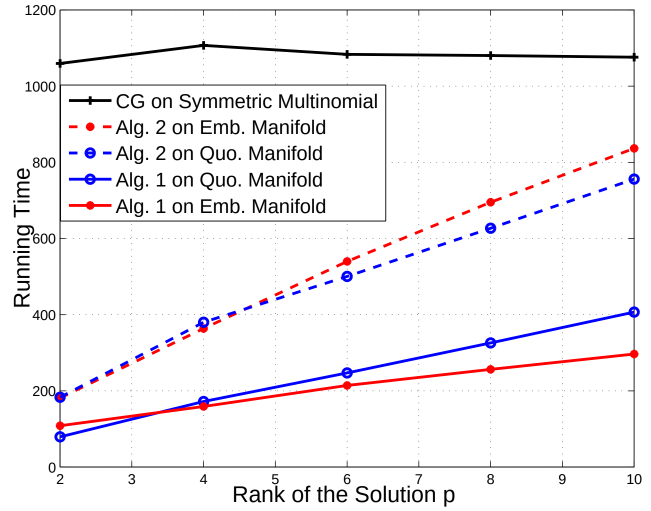


Figure 4.6: Performance of Algorithm 4.1 and Algorithm 4.2.

the first part of the simulation confirms the linear and quadratic convergence behavior of the proposed first and second-order methods, respectively. The second part runs Algorithm 4.1 and Algorithm 4.2 in order to find the optimal solution without any prior knowledge of the rank. The convergence is attested by checking that the solution is rank (almost) deficient from Theorem 4.2 which is cheaper than checking the first-order conditions in Theorem 4.1.

Figure 4.4 illustrates the convergence rate of the conjugate gradient method (first-order method) on the proposed embedded and quotient manifolds. The figure reveals that the proposed first-order methods exhibit a linear behavior as anticipated. One can note that the linear response of these methods mimics the one of unconstrained optimization as the Riemannian method transforms a constrained optimization into an unconstrained one. Similarly, Figure 4.5 plots the convergence rate of the second-order methods. These second-order methods achieve a superlinear convergence as compared with the first-order methods. The figure also illustrates that the quadratic behavior is reached after a certain number of iterations which can be explained by the fact that the quadratic approximation of the objective function in the TR method is tight only in a region close to the optimal value.

Figure 4.6 shows the running time of the proposed algorithms in solving the above optimization problem with unknown solution rank p. Recall that the symmetric multinomial solves the problem over the $\frac{n(n-1)}{2}$ space, unlike our method that reduces the dimension to $n(p-1)$. However, the geometry of the embedded and quotient manifold is more complex than the one of the symmetric multinational. Furthermore, as the rank of the solution is unknown, Algorithm 4.1 and Algorithm 4.2 solve the optimization multiple times with increasing rank. Nevertheless, Figure 4.6 attests that for a reasonable number of clusters p, our proposed Algorithm 4.1 and Algorithm 4.2 both perform better than the symmetric multinomial.

4.5 Beyond the Positive Multinomial Manifold

Several combinatorial applications of late can be relaxed as convex optimization problems with theoretical guarantees under which the solution to the convex relaxation coincides with the optimal solution under suitable conditions. A particularly interesting class of problems requires the optimization over stochastic or doubly stochastic matrices such as graph-based and spectral clustering [55, 53, 39, 109, 54], clustering by tensor decomposition [32], hidden Markov models [110], and graph optimization [111].

The previous sections took advantage from the fact the solution to these problem is low-rank to propose highly efficient algorithms. This part is interested in solving similar optimization problems while alleviating the restriction that the matrix is symmetric. In particular, by explicitly factorizing the $n \times m$ variable $\mathbf{X} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$ with $\mathbf{U}$ and $\mathbf{V}$ being $n \times p$ and $m \times p$ matrices, a significant complexity gain can be achieved for $p \ll \min(n, m)$, e.g., [98, 97]. However, as before, the factorization makes the problem non-convex with non-isolated solutions. Fortunately, the quality of the solution of the original convex problem can be theoretically attested, e.g., [102]. More recently, inspired by the Burer-Monteiro approach [99], the authors in

[112] showed that a large class of Semidefinite Programs (SDPs) almost never has any spurious local optima.

The study of the non-symmetric factorization is crucial to multiple non-symmetric clustering algorithms including the non-negative matrix factorization (NMF) [113, 96, 114, 115] of interest in this section. The rest of this section explicitly uses the low-rank property of the solution to reformulate the problem and investigate the geometry of the resulting manifold.

Problems and Manifolds of Interest

Let $\mathsf{f} : \mathbb{R}^{n\times n} \rightarrow \mathbb{R}$ be a smooth function and consider the following optimization problem

$$\min_{\mathbf{X}\in\mathbb{R}^{n\times n}} \mathsf{f}(\mathbf{X}) \tag{4.31a}$$

$$\text{s.}t.\ \mathbf{X} \geq \mathbf{0}, \tag{4.31b}$$

$$\mathbf{X}\mathbf{1} = \mathbf{1}, \tag{4.31c}$$

wherein constraint (4.31b) underlines that the matrix is *element-wise* positive and constraint (4.31c) corresponds to the fact that each row sums to 1, i.e., the matrix is stochastic.

This section factorizes the optimization variable to include the low-rank property in the structure of the variable. The paper assumes that the rank p of the solution is known a priori. Otherwise, one can solve the reformulated problems with increasing rank until a similar condition as the one in the previous section is satisfied. A matrix $\mathbf{X}$ of rank p can be factorized in various ways [116], e.g., full-rank factorization, polar decomposition, SVD, etc. The full-rank factorization $\mathbf{X} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$ with $\mathbf{U}$ and $\mathbf{V}$ being $n \times p$ full-rank matrices, denoted by $\mathbf{U}, \mathbf{V} \in \mathbb{R}_*^{n\times p}$, is adopted herein [101, 102, 117]. The reformulation allows reducing the dimension of the ambient space from n^2 to $2np$ with $p \ll n$.

Given the low-rank factorization $\mathbf{X} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$ and an invertible matrix $\mathbf{M}$, we have $(\mathbf{U}\mathbf{M}^{-1}, \mathbf{V}\mathbf{M}^{\mathrm{T}})$ represents the same factorization. Hence, one can study the quotient structure of the manifolds by the general linear group. However, as non-isolated solutions only impair second-order methods, the study of the quotient structure is left for future investigation and this thesis only focuses on the first-order geometries of the low-rank multinomial and the low-rank doubly stochastic manifolds defined as follows:

$$\mathcal{P}_n^p = \left\{(\mathbf{U},\mathbf{V}) \in \left(\mathbb{R}_*^{n\times p}\right)^2 \;\middle|\; \mathbf{U}\mathbf{V}^\mathrm{T} > \mathbf{0} \text{ and } \mathbf{U}\mathbf{V}^\mathrm{T}\mathbf{1} = \mathbf{1}\right\} \tag{4.32}$$

$$\mathcal{DP}_n^p = \left\{(\mathbf{U},\mathbf{V}) \in \left(\mathbb{R}_*^{n\times p}\right)^2 \;\middle|\; \mathbf{U}\mathbf{V}^\mathrm{T} > \mathbf{0}, \mathbf{U}\mathbf{V}^\mathrm{T}\mathbf{1} = \mathbf{1}, \text{ and } \mathbf{V}\mathbf{U}^\mathrm{T}\mathbf{1} = \mathbf{1}\right\}. \tag{4.33}$$

Geometry of the Low-Rank Multinomial

The low-rank multinomial manifold $\mathcal{P}_n^p$ is seen as an embedded manifolds of the product of non-compact Stiefel manifolds. The ambient space is of dimension $2np$ and the manifold is characterized by n linearly independent equations. Therefore, the manifold is of dimension $n(2p-1)$. The following proposition characterizes the tangent space.

Proposition 4.3 *The tangent space $\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p$ of the low-rank multinomial manifold $\mathcal{P}_n^p$ at the point $(\mathbf{U},\mathbf{V}) \in \mathcal{P}_n^p$ is given by*

$$\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p = \left\{(\xi_\mathbf{U},\eta_\mathbf{V}) \in \left(\mathbb{R}^{n\times p}\right)^2 \;\middle|\; (\xi_\mathbf{U}\mathbf{V}^\mathrm{T} + \mathbf{U}\eta_\mathbf{V}^\mathrm{T})\mathbf{1} = \mathbf{0}\right\}. \tag{4.34}$$

Proof: The proof of this proposition relies on showing the inclusion of the tangent space in the set illustrated in (4.34). Afterward, a dimension counting argument concludes the proof.

Consider the curve $\gamma(t) \in \mathcal{P}_n^p$ for $t \in \mathcal{I}$ with I being an open interval containing 0 going through $(\mathbf{U},\mathbf{V})$ at the origin, i.e., $\gamma(0) = (\mathbf{U},\mathbf{V})$. From the manifold description in (4.32), $\gamma(t) \in \mathcal{P}_n^p$ implies that $\gamma_1(t)\gamma_2(t)^\mathrm{T}\mathbf{1} = \mathbf{1}$. Differentiating the equation gives $(\dot{\gamma_1}(t)\gamma_2(t)^\mathrm{T} + \gamma_1(t)\dot{\gamma_2}(t)^\mathrm{T})\mathbf{1} = \mathbf{0}$ which evaluated at $t = 0$ gives the following inclusion

$$\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p \subseteq \left\{(\xi_\mathbf{U},\eta_\mathbf{V}) \in \left(\mathbb{R}^{n\times p}\right)^2 \;\middle|\; (\xi_\mathbf{U}\mathbf{V}^\mathrm{T} + \mathbf{U}\eta_\mathbf{V}^\mathrm{T})\mathbf{1} = \mathbf{0}\right\}.$$

Finally, notice that the dimension of the set on the right-hand side of (4.34) is $n(2p-1)$ which matches the dimension of the manifold. Therefore, we have equality of both sets which gives a parameterization of the tangent space. ■

This section endows the tangent space $\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p$ with the following Riemannian metric in anticipation that it is a compatible Riemannian metric with the quotient manifold (not included herein):

$$\langle(\xi_\mathbf{U},\eta_\mathbf{V}),(\xi'_\mathbf{U},\eta'_\mathbf{V})\rangle_{(\mathbf{U},\mathbf{V})} = \mathrm{Tr}((\mathbf{U}^\mathrm{T}\mathbf{U})^{-1}\xi_\mathbf{U}^\mathrm{T}\xi'_\mathbf{U}) + \mathrm{Tr}((\mathbf{V}^\mathrm{T}\mathbf{V})^{-1}\eta_\mathbf{V}^\mathrm{T}\eta'_\mathbf{V}). \tag{4.35}$$

The following lemma relates the Euclidean and Riemannian gradients.

Lemma 4.3 *The Riemannian gradient* grad $\mathsf{f}(\mathbf{U},\mathbf{V})$ *can be obtained from its Euclidean counterpart* Grad $\mathsf{f}(\mathbf{U},\mathbf{V})$ *using the following identity*

$$\text{grad } \mathsf{f}(\mathbf{U},\mathbf{V}) = \begin{pmatrix} \text{grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V}) \\ \text{grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V}) \end{pmatrix} = \Pi_{(\mathbf{U},\mathbf{V})} \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U}) \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V}) \end{pmatrix},$$

with $\Pi_{(\mathbf{U},\mathbf{V})}$ *being the orthogonal projection from the ambient space* $\left(\mathbb{R}^{n\times p}\right)^2$ *onto the tangent space* $\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p$.

Proof: Let $(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) \in \mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p$ be a tangent vector. The Euclidean gradient can be expressed as a directional derivative as follows:

$$\begin{aligned} \mathrm{D}\,\mathsf{f}(\mathbf{U},\mathbf{V})[(\xi_{\mathbf{U}},\eta_{\mathbf{V}})] &= \left\langle \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V}) \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V}) \end{pmatrix}, \begin{pmatrix} \xi_{\mathbf{U}} \\ \eta_{\mathbf{V}} \end{pmatrix} \right\rangle \\ &= \left\langle \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U}) \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V}) \end{pmatrix}, \begin{pmatrix} \xi_{\mathbf{U}} \\ \eta_{\mathbf{V}} \end{pmatrix} \right\rangle_{(\mathbf{U},\mathbf{V})} \\ &= \left\langle \Pi_{(\mathbf{U},\mathbf{V})} \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U}) \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V}) \end{pmatrix}, \begin{pmatrix} \xi_{\mathbf{U}} \\ \eta_{\mathbf{V}} \end{pmatrix} \right\rangle_{(\mathbf{U},\mathbf{V})}. \end{aligned}$$

Therefore, for all tangent vectors $(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) \in \mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}_n^p$, the tangent vector defined by $\Pi_{(\mathbf{U},\mathbf{V})} \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U}) \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V}) \end{pmatrix}$ satisfies the equation of Riemannian gradient which concludes the proof. ■

Given the identity in Lemma 4.3, the Riemannian gradient expression is provided in Theorem 4.5

Theorem 4.5 *The Riemannian gradient on the low-rank multinomial can be expressed as*

$$\begin{pmatrix} \text{grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V}) \\ \text{grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V}) \end{pmatrix} = \begin{pmatrix} \text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{U}^{\mathrm{T}}\mathbf{U} - \alpha\mathbf{1}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U} \\ \text{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{V}^{\mathrm{T}}\mathbf{V} - \mathbf{1}\alpha^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V} \end{pmatrix},$$

with α being the n-dimensional vector given by

$$\alpha = \frac{1}{n}(\mathbf{I} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}})^{-1}\left(\text{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\text{Grad}_{\mathbf{V}}^{\mathrm{T}}\mathsf{f}(\mathbf{U},\mathbf{V})\right)\mathbf{1}.$$

Proof: As shown previously, it is sufficient to derive the expression of the orthogonal projection onto the tangent space to get the expression of the Riemannian gradient. First, notice that the orthogonal complement of the tangent space is characterized by an n-dimensional vector α as follows

$$\mathcal{T}^{\perp}_{(\mathbf{U},\mathbf{V})}\mathcal{P}^p_n = \left\{(\mathbf{A},\mathbf{B}) \in \left(\mathbb{R}^{n\times p}\right)^2 \;\middle|\; \mathbf{A} = \alpha\mathbf{1}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U} \text{ and } \mathbf{B} = \mathbf{1}\alpha^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\right\}.$$

Let $(\mathbf{R},\mathbf{S})$ be an ambient vector and decompose it into a tangent and an orthogonal components, i.e., $(\mathbf{R},\mathbf{S}) = \Pi_{(\mathbf{U},\mathbf{V})}(\mathbf{R},\mathbf{S}) + \Pi^{\perp}_{(\mathbf{U},\mathbf{V})}(\mathbf{R},\mathbf{S})$. Using the definition of the tangent space in Proposition 4.3 and the parameterization of the orthogonal complement to the tangent space, we obtain the following equation

$$(\mathbf{R}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{S}^{\mathrm{T}})\mathbf{1} = (\mathbf{A}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{B}^{\mathrm{T}})\mathbf{1} = (\alpha\mathbf{1}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}}\alpha\mathbf{1}^{\mathrm{T}})\mathbf{1}.$$

Using the fact that $\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{1} = \mathbf{1}$, the vector α can be expressed as $\alpha = \frac{1}{n}(\mathbf{I} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}})^{-1}(\mathbf{R}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{S}^{\mathrm{T}})\mathbf{1}$. Finally, applying the above result to matrices $\mathbf{R}$ and $\mathbf{S}$ equal to $\mathsf{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U})$ and $\mathsf{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V})$, respectively, concludes the proof. ■

This part describes a first-order retraction on the low-rank multinomial $\mathcal{P}^p_n$ and its positive extension $\overline{\mathcal{P}}^p_n = \left\{(\mathbf{U},\mathbf{V}) \in \left(\mathbb{R}^{n\times p}_*\right)^2 \;\middle|\; \mathbf{U} > \mathbf{0}, \mathbf{V} > \mathbf{0}, \text{ and } \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{1} = \mathbf{1}\right\}$. First, let $\Pi : \mathbb{R}^{n\times n}_+ \longrightarrow \mathcal{P}_n$ be the projection[1] of entry-wise positive matrices to the set of stochastic matrices $\mathcal{P}_n$ given by $\Pi(\mathbf{X}) = \mathbf{D}\mathbf{X}$ wherein $\mathbf{D}$ is a strictly positive diagonal matrix such that $\mathbf{D}_{ii} = \left(\sum_{j=1}^n \mathbf{X}_{ij}\right)^{-1}$. Extend the definition of Π to $\mathcal{P}^p_n$ such that $\Pi(\mathbf{U},\mathbf{V}) = (\mathbf{D}\mathbf{U},\mathbf{V})$ with $\mathbf{D}$ obtained by applying Π to the $n\times n$ entry-wise positive matrix $\mathbf{U}\mathbf{V}^{\mathrm{T}}$. The following theorem derives the expression of a first-order retraction

Theorem 4.6 *Consider the tangent space $\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}^p_n$ and the tangent vector $(\xi_{\mathbf{U}},\eta_{\mathbf{V}})$ and define $\mathbf{U}^{\xi} = \mathbf{U} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp(-\xi_{\mathbf{U}})$ and $\mathbf{V}^{\eta} = \mathbf{V} + \mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp(-\eta_{\mathbf{V}})$ for the usual exponential function. The operator $\mathsf{R}_{(\mathbf{U},\mathbf{V})} : \mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{P}^p_n \longrightarrow \mathcal{P}^p_n$ defined by*

$$\mathsf{R}_{(\mathbf{U},\mathbf{V})}(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) = \Pi(\mathbf{U}^{\xi},\mathbf{V}^{\eta}),$$

is a well-defined retraction for small-enough tangent vectors, i.e., for $(\xi_{\mathbf{U}},\eta_{\mathbf{V}})$ such that $\mathbf{U}^{\xi}\,(\mathbf{V}^{\eta})^{\mathrm{T}} \in \mathbb{R}^{n\times n}_+$.

Proof: It is sufficient to demonstrate the centering and the local rigidity properties to show that $\mathsf{R}_{(\mathbf{U},\mathbf{V})}$ is a first-order retraction. The centering properties can be directly obtained by replacing the tangent vectors by zeros. The study of the local rigidity is

[1] Not to be confused with $\Pi_{(\mathbf{U},\mathbf{V})}$, the orthogonal projection onto the tangent space.

accomplished by studying the first-order perturbation of $(\mathbf{U}^{t\xi}, \mathbf{V}^{t\eta})$ and the projection operator Π. Note that the first-order approximation of $\mathbf{U}^{t\xi} = \mathbf{U}+\mathbf{1}_n\mathbf{1}_p^{\mathrm{T}} - \exp\big(-t\xi_{\mathbf{U}}\big) = \mathbf{U} + t\xi_{\mathbf{U}} + o(t^2\mathbf{1}_n\mathbf{1}_p^{\mathrm{T}})$. The same holds for $\mathbf{V}^{t\eta}$. Therefore, it is sufficient to study the perturbation of Π in the direction of the tangent space. By definition, we have $\Pi(\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}}) = ((\mathbf{D} + \delta\mathbf{D})(\mathbf{U} + \xi_{\mathbf{U}}), \mathbf{V} + \eta_{\mathbf{V}})$. For $(\mathbf{U}, \mathbf{V}) \in \mathcal{P}_n^p$, the matrix $\mathbf{U}\mathbf{V}^{\mathrm{T}}$ is stochastic, i.e., $\mathbf{D} = \mathbf{I}$. Therefore, expanding the stochastic matrix given by $\Pi(\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}})$ and keeping only the first-order terms, we obtain

$$(\mathbf{I} + \delta\mathbf{D})(\mathbf{U} + \xi_{\mathbf{U}})(\mathbf{V} + \eta_{\mathbf{V}})^{\mathrm{T}} = \mathbf{U}\mathbf{V}^{\mathrm{T}} + \delta\mathbf{D}\mathbf{U}\mathbf{V}^{\mathrm{T}} + \xi_{\mathbf{U}}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\eta_{\mathbf{V}}^{\mathrm{T}}.$$

Since the matrix is stochastic, then by multiplying the above equality by the all ones vector we obtain

$$\mathbf{1} = \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{1} + \delta\mathbf{D}\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{1} + (\xi_{\mathbf{U}}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\eta_{\mathbf{V}}^{\mathrm{T}})\mathbf{1} = \mathbf{1} + \delta\mathbf{D}\mathbf{1} \Rightarrow \delta\mathbf{D}\mathbf{1} = \mathbf{0} \Rightarrow \delta\mathbf{D} = \mathbf{0}.$$

By showing above that $\Pi(\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}}) = (\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}})$, one obtains the first-order approximation of $\mathsf{R}_{(\mathbf{U},\mathbf{V})}$ as

$$\begin{aligned} \mathsf{R}_{(\mathbf{U},\mathbf{V})}(t\xi_{\mathbf{U}}, t\eta_{\mathbf{V}}) &= \Pi(\mathbf{U}^{t\xi}, \mathbf{V}^{t\eta}) = \Pi(\mathbf{U} + t\xi_{\mathbf{U}}, \mathbf{V} + t\eta_{\mathbf{V}}) + o(t^2\mathbf{1}_n\mathbf{1}_p^{\mathrm{T}}) \\ &= (\mathbf{U} + t\xi_{\mathbf{U}}, \mathbf{V} + t\eta_{\mathbf{V}}) + o(t^2\mathbf{1}_n\mathbf{1}_p^{\mathrm{T}}), \end{aligned}$$

which shows the local rigidity property as desired. ■

Corollary 4.1 *A first-order retraction on the positive low-rank multinomial of* any *tangent vector* $(\xi_{\mathbf{U}}, \eta_{\mathbf{V}}) \in \mathcal{T}_{(\mathbf{U},\mathbf{V})}\overline{\mathcal{P}}_n^p$ *is given by*

$$\mathsf{R}_{(\mathbf{U},\mathbf{V})}(\xi_{\mathbf{U}}, \eta_{\mathbf{V}}) = \Pi(\overline{\mathbf{U}}^{\xi}, \overline{\mathbf{V}}^{\eta}),$$

with $\overline{\mathbf{U}}_{ij}^{\xi} = \mathbf{U}_{ij}\exp\big(\xi_{ij}/\mathbf{U}_{ij}\big)$ *and* $\overline{\mathbf{V}}_{ij}^{\eta} = \mathbf{V}_{ij}\exp\big(\eta_{ij}/\mathbf{V}_{ij}\big)$.

Proof: The proof of this corollary is straightforward by noticing that the first-order approximation of $\mathbf{Z}_{ij} = \mathbf{X}_{ij}\exp\big(\xi_{ij}/\mathbf{X}_{ij}\big)$ is given by $\mathbf{X}_{ij} + \xi_{ij}$ and using similar steps as in the proof of Theorem 4.6. ■

The Low-Rank Doubly Stochastic Manifold

Similar to the low-rank multinomial manifold, the low-rank doubly stochastic manifold $\mathcal{DP}_n^p = \left\{(\mathbf{U}, \mathbf{V}) \in \left(\mathbb{R}_*^{n\times p}\right)^2 \;\middle|\; \mathbf{U}\mathbf{V}^{\mathrm{T}} > \mathbf{0}, \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{1} = \mathbf{1}, \text{ and } \mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{1} = \mathbf{1}\right\}$ is seen an embedded manifold of the product of non-compact Stiefel manifolds. The dimension of the manifold is $2n(p-1)+1$ which can be obtained from the Birkhoff–Von Neumann theorem [118].

Corollary 4.2 *The tangent space for the low-rank doubly stochastic manifold is characterized by*

$$\mathcal{T}_{(\mathbf{U},\mathbf{V})}\mathcal{DP}_n^p = \left\{(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) \in \left(\mathbb{R}^{n\times p}\right)^2 \;\middle|\; (\xi_{\mathbf{U}}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\eta_{\mathbf{V}}^{\mathrm{T}})\mathbf{1} = (\mathbf{V}\xi_{\mathbf{U}}^{\mathrm{T}} + \eta_{\mathbf{V}}\mathbf{U}^{\mathrm{T}})\mathbf{1} = \mathbf{0}\right\}. \tag{4.36}$$

Proof: The proof of this theorem is omitted as it mirrors the steps used in proving Proposition 4.3. ■

Using the Riemannian inner product as defined in (4.35), the orthogonal complement to the tangent space is the $2n-1$-dimensional space parametrized by two $n\times 1$ vectors α and β as follows

$$\begin{aligned}\mathcal{T}^{\perp}_{(\mathbf{U},\mathbf{V})}\mathcal{DP}_n^p = \Big\{(\mathbf{A},\mathbf{B}) \in \mathbb{R}^{n\times p} \;\Big|\; & \mathbf{A} = (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}})\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U}, \\ & \mathbf{B} = (\mathbf{1}\alpha^{\mathrm{T}} + \beta\mathbf{1}^{\mathrm{T}})\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\Big\}.\end{aligned}$$

As both the low-rank multinomial and the low-rank doubly stochastic manifolds share the same Riemannian metric, the relationship between the Riemannian gradient and its Euclidean counterpart provided in Lemma 4.3 is still valid. Therefore, one only needs to derive an expression of the orthogonal projection onto the tangent space to get the expression of the gradient as shown in the following corollary

Corollary 4.3 *The Riemannian gradient on the low-rank multinomial can be expressed as*

$$\begin{pmatrix}\mathsf{grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})\\ \mathsf{grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})\end{pmatrix} = \begin{pmatrix}\mathsf{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{U}^{\mathrm{T}}\mathbf{U} - (\alpha\mathbf{1}^{\mathrm{T}} + \mathbf{1}\beta^{\mathrm{T}})\mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U}\\ \mathsf{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{V}^{\mathrm{T}}\mathbf{V} - (\mathbf{1}\alpha^{\mathrm{T}} + \beta\mathbf{1}^{\mathrm{T}})\mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\end{pmatrix}. \tag{4.37}$$

Define $\mathbf{M}_1 = (\mathbf{I} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}})$ *and* $\mathbf{M}_2 = (\mathbf{I} + \mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}})$*, the vectors* α *and* β *are* a solution *to the over-determined system of equations*

$$\begin{pmatrix} n\mathbf{M}_1 & \mathbf{M}_1\mathbf{1}\mathbf{1}^{\mathrm{T}}\\ \mathbf{M}_2\mathbf{1}\mathbf{1}^{\mathrm{T}} & n\mathbf{M}_2\end{pmatrix}\begin{pmatrix}\alpha\\ \beta\end{pmatrix} = \begin{pmatrix}\mathsf{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathsf{Grad}_{\mathbf{V}}^{\mathrm{T}}\mathsf{f}(\mathbf{U},\mathbf{V})\\ \mathsf{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})\mathbf{V}^{\mathrm{T}}\mathbf{V}\mathbf{U}^{\mathrm{T}} + \mathbf{V}\mathbf{U}^{\mathrm{T}}\mathbf{U}\mathsf{Grad}_{\mathbf{U}}^{\mathrm{T}}\mathsf{f}(\mathbf{U},\mathbf{V})\end{pmatrix}\begin{pmatrix}\mathbf{1}\\ \mathbf{1}\end{pmatrix}.$$

Proof: The proof of this corollary is similar to the proof of Theorem 4.5. As such and due to space limitations, the details of the computation are greatly compressed. Let $(\mathbf{R},\mathbf{S})$ be an ambient vector and decompose it, as before, into a tangent and an orthogonal component. Recall that tangent vectors satisfy $(\xi_{\mathbf{U}}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\eta_{\mathbf{V}}^{\mathrm{T}})\mathbf{1} =$

$(\mathbf{V}\xi_{\mathbf{U}}^{\mathrm{T}} + \eta_{\mathbf{V}}\mathbf{U}^{\mathrm{T}})\mathbf{1} = \mathbf{0}$. Therefore, exploiting the characterization of the orthogonal complement of the tangent space allows to write the following equations:

$$\begin{aligned}(\mathbf{R}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{S}^{\mathrm{T}})\mathbf{1} &= (\mathbf{A}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{B}^{\mathrm{T}})\mathbf{1}\\ (\mathbf{V}\mathbf{R}^{\mathrm{T}} + \mathbf{S}\mathbf{U}^{\mathrm{T}})\mathbf{1} &= (\mathbf{V}\mathbf{A}^{\mathrm{T}} + \mathbf{B}\mathbf{U}^{\mathrm{T}})\mathbf{1}.\end{aligned}$$

Solving the above system of equations in the unknown α and β and substituting $\mathbf{R}$ and $\mathbf{S}$ by $\mathsf{Grad}_{\mathbf{U}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{U}^{\mathrm{T}}\mathbf{U})$ and $\mathsf{Grad}_{\mathbf{V}}\mathsf{f}(\mathbf{U},\mathbf{V})(\mathbf{V}^{\mathrm{T}}\mathbf{V})$, respectively, gives the expression for the Riemannian gradient illustrated in (4.37). ■

This part extends the proposed retractions for the low-rank doubly stochastic manifold $\mathcal{DP}_n^p$ and its positive extension $\overline{\mathcal{DP}}_n^p$. Let $\Pi : \mathbb{R}_+^{n\times n} \longrightarrow \mathcal{DP}_n$ be the projection of entry-wise positive matrices to the set of doubly stochastic matrices given by $\Pi(\mathbf{X}) = \mathbf{D}_1\mathbf{X}\mathbf{D}_2$ wherein $\mathbf{D}_1$ and $\mathbf{D}_2$ are strictly positive diagonal matrix obtained from the Sinkhorn-Knopp algorithm [84]. As before, extend the definition of Π to $\mathcal{DP}_n^p$ such that $\Pi(\mathbf{U},\mathbf{V}) = (\mathbf{D}_1\mathbf{U},\mathbf{D}_2\mathbf{V})$ with $\mathbf{D}_1$ and $\mathbf{D}_2$ being obtained by applying Π to the $n \times n$ entry-wise positive matrix $\mathbf{U}\mathbf{V}^{\mathrm{T}}$. The following theorem derives the expression of a first-order retraction on $\mathcal{DP}_n^p$ and $\overline{\mathcal{DP}}_n^p$

Theorem 4.7 *For small-enough tangent vectors, an efficient first-order retraction on the low-rank doubly stochastic manifold $\mathcal{DP}_n^p$ is given by $\mathsf{R}_{(\mathbf{U},\mathbf{V})}(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) = \Pi(\mathbf{U}^{\xi},\mathbf{V}^{\eta})$, for $\mathbf{U}^{\xi}$ and $\mathbf{V}^{\eta}$ as defined in Theorem 4.6. Similarly, for $\overline{\mathbf{U}}^{\xi}$ and $\overline{\mathbf{V}}^{\eta}$ given in Corollary 4.1, a retraction on $\overline{\mathcal{DP}}_n^p$ can be defined by $\mathsf{R}_{(\mathbf{U},\mathbf{V})}(\xi_{\mathbf{U}},\eta_{\mathbf{V}}) = \Pi(\overline{\mathbf{U}}^{\xi},\overline{\mathbf{V}}^{\eta})$.*

Proof: The proof of this theorem follows similar steps as the one for the low-rank multinomial with the exception of the projection operator Π. Therefore, for conciseness purposes, this proof only focuses on the study of the first-order perturbation of Π in the direction of tangent vector. Recall that $\Pi(\mathbf{U},\mathbf{V}) = (\mathbf{D}_1\mathbf{U},\mathbf{D}_2\mathbf{V})$ with $\mathbf{D}_1$ and $\mathbf{D}_2$ being obtained from the Sinkhorn-Knopp algorithm. Given the tangent perturbation $(\xi_{\mathbf{U}},\eta_{\mathbf{V}})$, the first-order perturbation of the projection is expressed as

$$\begin{aligned}\Pi(\mathbf{U}+\xi_{\mathbf{U}},\mathbf{V}+\eta_{\mathbf{V}}) &= (\mathbf{D}_1+\delta\mathbf{D}_1)(\mathbf{U}+\xi_{\mathbf{U}})(\mathbf{V}^{\mathrm{T}}+\eta_{\mathbf{V}}^{\mathrm{T}})(\mathbf{D}_2+\delta\mathbf{D}_2)\\ &= \mathbf{U}\mathbf{V}^{\mathrm{T}} + \delta\mathbf{D}_1\mathbf{U}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\mathbf{V}^{\mathrm{T}}\delta\mathbf{D}_2 + (\xi_{\mathbf{U}}\mathbf{V}^{\mathrm{T}} + \mathbf{U}\eta_{\mathbf{V}}^{\mathrm{T}}),\end{aligned}$$

wherein the last equality is obtained by keeping only the first-order terms and substituting $\mathbf{D}_1$ and $\mathbf{D}_2$ by identities since $\mathbf{U}\mathbf{V}^{\mathrm{T}}$ is a doubly stochastic matrix. Define $\delta\mathbf{D}_1 = \delta\mathbf{D}_1\mathbf{1}$ and $\delta\mathbf{D}_2 = \delta\mathbf{D}_2\mathbf{1}$. Then, by multiplying the above equation on the right and left by $\mathbf{1}$ and $\mathbf{1}^{\mathrm{T}}$, respectively, and using the characterization of tangent vectors, we obtain the following matrix equations

$$\begin{pmatrix} \mathbf{I} & \mathbf{U}\mathbf{V}^{\mathrm{T}} \\ \mathbf{V}\mathbf{U}^{\mathrm{T}} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \delta\mathbf{D}_1 \\ \delta\mathbf{D}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}.$$

In other words, $\begin{pmatrix} \delta\mathbf{D}_1 \\ \delta\mathbf{D}_2 \end{pmatrix}$ is an eigenvalue associated with the eigenvalue 0. A similar analysis to [75] allows to show that $\Pi(\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}}) = (\mathbf{U} + \xi_{\mathbf{U}}, \mathbf{V} + \eta_{\mathbf{V}})$ which concludes the proof. ■

Numerical Results

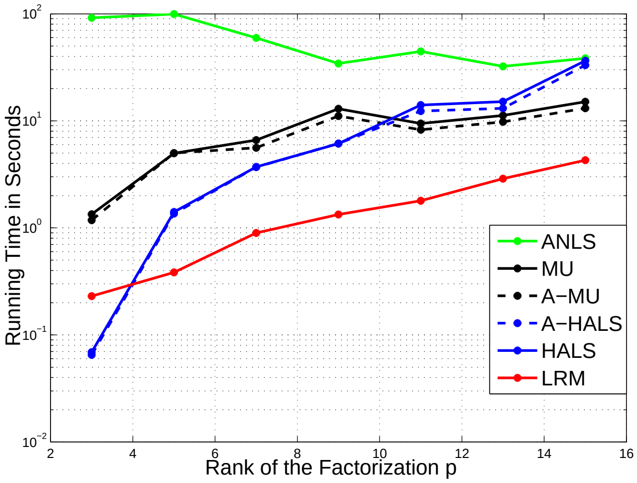
This subsection attests the performance of the proposed manifold optimization algorithms for non-negative matrix factorization applications using real-world and synthetic data. In particular, the section compares the performance of the low-rank multinomial (denoted by LRM) against popular[2] NMF algorithms [114] and their accelerated variants [120] namely

- Alternating Least-Squares (ALS).
- Alternating Non-negative Least-Squares (ANLS).
- Hierarchical Alternating Least-Squares (HALS).
- Accelerated HALS (A-HALS).
- The Multiplicative Updates (MU).
- Accelerated MU (A-MU).

As stated in above, given the above derived description of the first-order geometry, optimization algorithms can be obtained mechanically using the MATLAB manifold optimization toolbox Manopt [37]. The numerical results herein are obtained using the conjugate gradient (CG) algorithm on the low-rank multinomial. Unlike the steepest-descent method, CG requires a vector transport $\mathcal{T}_{\eta_{\mathbf{X}}}\xi_{\mathbf{X}}$ which can be computed [23] for matrix manifolds as $\mathcal{T}_{\eta_{\mathbf{X}}}\xi_{\mathbf{X}} = \Pi_{\mathsf{R}_{\mathbf{X}}(\eta_{\mathbf{X}})}(\xi_{\mathbf{X}})$.

Given an $n \times m$ matrix $\mathbf{A}$ generated as $\mathbf{A} = \mathbf{U}\mathbf{V}^{\mathrm{T}} + \mathbf{W}$, wherein $\mathbf{U}$ and $\mathbf{V}$ are two element-wise non-negative matrices of size $n \times p$ and $m \times p$, respectively, and $\mathbf{W}$

[2] The Projected Gradient Methods (PGM) and its accelerated (A-PGM) version [119] are not tested herein due to their poor performance on our simulation set.



$\frac{\|\|\mathbf{A}-\mathbf{U}\mathbf{V}^{\mathrm{T}}\|\|_F}{\|\|\mathbf{A}\|\|_F}$	$p = 7$	$p = 9$	$p = 11$
LRM	0.6851	0.6533	0.6222
ALS	0.8330	0.8043	0.7777
ANLS	0.6835	0.6488	0.6143
MU/A-MU	0.6540	0.6228	0.5928
HALS/A-HALS	0.6534	0.6224	0.5920

Figure 4.7: Performance of the different NMF algorithms.

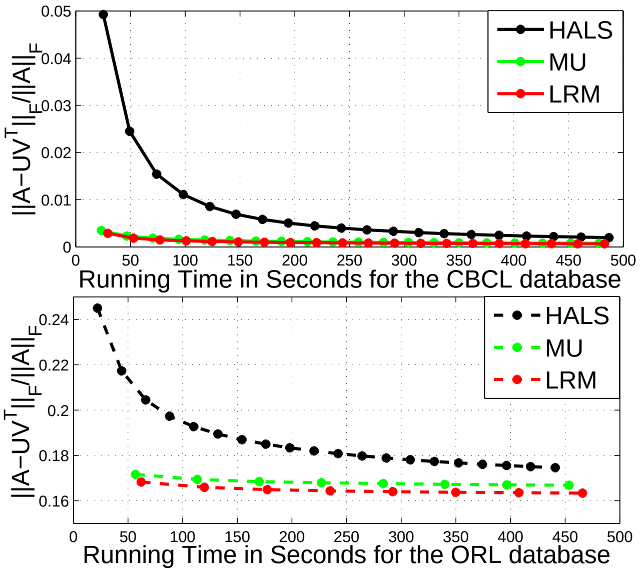


Figure 4.8: Running time to decompose the ORL and the CBCL face databases.

is a white Gaussian noise such that the signal-to-noise ratio (SNR) is 20dB. This section proposes factorizing $\mathbf{A}$ by solving the problem $\min_{(\mathbf{U},\mathbf{V})\in\overline{\mathcal{P}}^{p}_{n,m}} ||\mathbf{A}-\mathbf{D}\mathbf{U}\mathbf{V}^{\mathrm{T}}||_F^2$ wherein $\mathbf{D}$ is a diagonal matrix containing the sum of each row of $\mathbf{A}$. In other words, the above program insists on finding a non-negative factorization close to $\mathbf{A}$ that preserves the row sums. The factors $\mathbf{DU}$ and $\mathbf{V}$ are then used to initialize the HALS and MU algorithms, respectively, in Figure 4.7 and Figure 4.8, in order to obtain the non-negative matrix factorization of $\mathbf{A}$. All algorithms are initialized with the ALS solution, i.e., the positive part of the low-rank approximation of $\mathbf{A}$, and stopped as soon as the cost drops below the one of the LRM or the number of iterations exceeds the threshold of $N = 10^5$ iterations.

Figure 4.7 illustrates the performance and the running time of the different NMF algorithms for recovering a synthetically generated 50×80 matrix with rank p varying from 3 to 15. The figure clearly shows that using the proposed method and applying the HALS algorithms provides quicker convergence then directly applying the MU, HALS, or their accelerated versions. In other words, the proposed method presents the best performance for the considered scenario, i.e., $p \ll \min(n, m)$.

Figure 4.8 shows a similar comparison for real-world data using the CBCL [121] and the ORL [122] face database which contains 2429 and 400 gray-level 19×19 and 92×112 pixels images, respectively. Similar to [123, 122], the rank of the decomposition is set to $p = 49$ and $p = 30$, respectively. Due to its high computation, the ANLS algorithm is not simulated for this scenario. In addition, HALS (MU) and A-HALS (A-MU) have similar performance. The proposed algorithm significantly improves over HALS, in terms of running time, and slightly outperforms MU on the CBLC and ORL databases as the assumption $p \ll \min(n, m)$ is hardly met unlike in the synthetic data simulation in Figure 4.7.

Chapter 5

FAST FOURIER PHASE RETRIEVAL THROUGH MANIFOLD OPTIMIZATION

[1] A. Douik, F. Salehi, and B. Hassibi. “A Novel Riemannian Optimization Approach and Algorithm for Solving the Phase Retrieval Problem”. In: *Proc. of the 53rd Asilomar Conference on Signals, Systems, and Computers (Asilomar’ 2019), Asilomar, CA, USA*. Vol. 1. 1. Nov. 2019, pp. 1962–1966. DOI: 10.1109/IEEECONF44664.2019.9049040.

Several imaging applications require constructing the phase of a complex signal given observations of its amplitude. In most applications, a subset of phaseless measurements, say the discrete Fourier transform of the signal, form an orthonormal basis that can be exploited to speed up the recovery. This chapter suggests a novel Riemannian optimization approach for solving the Fourier phase retrieval problem by studying and exploiting the geometry of the problem to reduce the ambient dimension and derive extremely fast and accurate algorithms. The phase retrieval problem is reformulated as a constrained problem and a novel Riemannian manifold, referred to as the *fixed norms* manifold, is introduced to represent all feasible solutions. The geometry of the Riemannian manifold is derived in closed form which allows the design of highly efficient optimization algorithms. Furthermore, the quotient structure of the manifold is investigated to further accelerate the optimization. Numerical simulations indicate that the proposed approach outperforms conventional optimization-based methods both in accuracy and in convergence speed. The results presented herein are available in the research paper [44] and as such some of the text appears as it is in the publication.

5.1 The Phase Retrieval Problem

Overview of the State-of-the-Art Methods

The phase retrieval problem is a classical problem [42] in which one is interested in recovering an n-dimensional complex signal from m observations in the form of the amplitude of its linear combinations. It has a rich history, in terms of both theoretical results (see, e.g. [124, 56, 43] and references therein), and practical algorithms [125, 126, 127], that spans over generations of researchers thanks to its numerous applications in engineering and applied sciences such as imaging [128], crystallography and optics [129]. Indeed, due to the inability of physical measurement devices to detect phases, e.g., a photosensitive film that measures the light intensity, only the magnitude measurements are feasible in such applications.

In many applications of the phase retrieval problem, a subset of phaseless measurements is obtained via an orthonormal basis. For example, in the Fourier phase retrieval problem the goal is to reconstruct the signal from phaseless measurements of its discrete Fourier transform (DFT) [57, 58, 59]. This particular structure of the phase retrieval problem allows its reformulation as a constrained optimization problem wherein the constraint set is represented by an orthonormal basis. This chapter suggests exploiting the problem structure to reduce the dimension of the problem and design fast recovery algorithms using Riemannian optimization techniques. To the best of the author's knowledge, the fixed norms manifold has not been introduced nor studied in the literature, except for the simple case of a single observation in which the fixed norms manifold coincides with the complex sphere in $\mathbb{C}^n$, $\mathcal{S}^{n-1}$.

Problem Formulation and Assumptions

Let $\mathbf{x}_0 \in \mathbb{C}^n$ be a complex vector of dimension n and assume that the m observations are obtained by $\sqrt{b_i} = |\mathbf{a}_i^\star \mathbf{x}_0|$, $1 \le i \le m$ with the sensing vectors $\mathbf{a}_i \in \mathbb{C}^n$. Consider a smooth loss function ℓ, the phase retrieval problem can be formulated as

$$\min_{\mathbf{x}\in\mathbb{C}^n} \sum_{i=1}^{m} \ell(|\mathbf{a}_i^\star \mathbf{x}|, \sqrt{b_i}) \tag{5.1}$$

Without loss of generality, assume that the first k observations are obtained from an orthogonal basis, say the discrete Fourier transform for $k = n$. In other words, matrices $\mathbf{A}_i = \mathbf{a}_i \mathbf{a}_i^\star$, $1 \le i \le k$ are orthogonal projection matrices that collectively span the whole ambient space $\mathbb{C}^n$, i.e., $\mathbf{A}_i = \mathbf{A}_i^\star$, $\mathbf{A}_i \mathbf{A}_j = \delta_{ij} \mathbf{A}_i$ and $\sum_{i=1}^{k} \mathbf{A}_i = \mathbf{I}_n$. As an example, in Fourier phase retrieval $k = n$, and $\mathbf{a}_i$, for $i = 1, 2, \ldots, n$, is the i^{th} row of the $n \times n$ DFT matrix.

Proposed Approach: The Fixed Norms Manifold

The unconstrained optimization of the phase retrieval problem in (5.1) can be formulated as a constrained optimization as follows:

$$\min_{\mathbf{x}\in\mathbb{C}^n} \sum_{i=k+1}^{m} \ell(|\mathbf{a}_i^\star \mathbf{x}|, \sqrt{b_i}) \tag{5.2a}$$

$$\text{s.}t. \; |\mathbf{a}_i^\star \mathbf{x}| = \sqrt{b_i}, \; 1 \le i \le k \; . \tag{5.2b}$$

For a complex number $c \in \mathbb{C}$, the notation $\mathrm{Re}(c) = \frac{1}{2}(c + c^\star)$ represents the real part of c. Likewise, the symbol $\mathrm{Im}(c) = \frac{1}{2\mathrm{i}}(c - c^\star)$ refers to the imaginary part of c wherein i denotes the imaginary unit, i.e., $\mathrm{i}^2 = -1$. Clearly, the modulus equality constraint $|\mathbf{a}_i^\star \mathbf{x}| = \sqrt{b_i}$ of (5.2) is equivalent the quadratic constraint $\mathbf{x}^\star \mathbf{a}_i \mathbf{a}_i^\star \mathbf{x} = b_i$. Define the matrices $\mathbf{A}_i = \mathbf{a}_i \mathbf{a}_i^\star,\ 1 \le i \le k$. From the previous assumptions on the system model, the set of matrices $\{\mathbf{A}_i\}_{i=1}^k$ are orthogonal projection matrices that collectively span the whole ambient space $\mathbb{C}^n$. In other words, for all $1 \le i, j \le k$, we have $\mathbf{A}_i = \mathbf{A}_i^\star$, $\mathbf{A}_i \mathbf{A}_j = \delta_{ij} \mathbf{A}_i$ and $\sum_{i=1}^k \mathbf{A}_i = \mathbf{I}_n$. Let $\mathcal{M}$ denote the set of solutions to the optimization problem (5.2), i.e., $\mathcal{M} = \left\{ \mathbf{x} \in \mathbb{C}^n \mid \mathbf{x}^\star \mathbf{A}_i \mathbf{x} = b_i,\ 1 \le i \le k \right\}$, called herein the *fixed norms* manifold. The optimization problem (5.2) can be reformulated as,

$$\min_{\mathbf{x} \in \mathcal{M}} \sum_{i=k+1}^{m} \ell(|\mathbf{a}_i^\star \mathbf{x}|, \sqrt{b_i}). \tag{5.3}$$

5.2 The Embedded Fixed Norms Manifold Geometry

This section investigates the first and second-order geometry of the fixed norms manifold seen as an embedded manifold in the Euclidean space $\mathbb{R}^n \times \mathbb{R}^n$ which is isomorphic to $\mathbb{C}^n$. We compute the linear approximation of the manifold at each point, known as the tangent space, and we further characterize its geodesics and provide an expression of the Exponential map. All these ingredients are used later on to derive efficient first-order optimization algorithms on the manifold to solve the phase retrieval problem.

Manifold, Tangent Space, and Projection

Given a set of k non-negative and orthogonal $n \times n$ projection matrices $\{\mathbf{A}_i\}_{i=1}^k$ over the complex field $\mathbb{C}$, i.e., $\mathbf{A}_i \succeq \mathbf{0}$ and $\mathbf{A}_i \mathbf{A}_j = \delta_{ij} \mathbf{A}_i$ for all $1 \le i, j \le k$, satisfying $\sum_{i=1}^k \mathbf{A}_i = \mathbf{I}_n$ and k positive real numbers $\{b_i\}_{i=1}^k \in \mathbb{R}_{++}$, the fixed norms manifold is defined by,

$$\mathcal{M} = \left\{ \mathbf{x} \in \mathbb{C}^n \mid \mathbf{x}^\star \mathbf{A}_i \mathbf{x} = b_i,\ 1 \le i \le k \right\} \ . \tag{5.4}$$

Theorem 5.1 *The set $\mathcal{M}$ is a well-defined real manifold of dimension $2n - k$ embedded in $\mathbb{R}^n \times \mathbb{R}^n$, which is isomorphic to $\mathbb{C}^n$, and whose tangent space at $\mathbf{x} \in \mathcal{M}$ is given by,*

$$\mathcal{T}_{\mathbf{x}}\mathcal{M} = \left\{ \xi_{\mathbf{x}} \in \mathbb{C}^n \mid \xi_{\mathbf{x}}^\star \mathbf{A}_i \mathbf{x} + \mathbf{x}^\star \mathbf{A}_i \xi_{\mathbf{x}} = 0, 1 \le i \le k \right\} \tag{5.5}$$

Proof: In order to show that the set $\mathcal{M}$ is a manifold with dimension $2n - k$ and compute the expression of its tangent spaces, we use the following special case of the constant-rank theorem known as the implicit function theorem

Theorem 5.2 *Let $f : \mathbb{R}^n \longrightarrow \mathbb{R}^m$ be a smooth constant-rank function near $\mathbf{0}$. Then, the set $\mathcal{M} = f^{-1}(\mathbf{0})$ is an embedded manifold of $\mathbb{R}^n$ with dimension $n - m$. Furthermore, the tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ is given by all directions $\xi_{\mathbf{x}}$ for which the directional derivative of f is zero, i.e., $\mathcal{T}_{\mathbf{x}}\mathcal{M} = \mathsf{Ker}\ (\mathsf{D}\ \mathsf{f}(\mathbf{x}))$.*

Define the function $\mathsf{f} : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}^k$ such that $\mathsf{f}(\mathbf{y}, \mathbf{z}) = (\mathbf{x}^\star \mathbf{A}_1 \mathbf{x} - b_1, \cdots, \mathbf{x}^\star \mathbf{A}_k \mathbf{x} - b_k)^T$ wherein $\mathbf{x}$ is defined as $\mathbf{x} = \mathbf{y} + i \cdot \mathbf{z}$. For simplicity, without loss of generality, the rest of the proof considers that f accepts a complex argument $\mathbf{x} \in \mathbb{C}^n$. It is not difficult to see that $\mathcal{M} = f^{-1}(\mathbf{0})$. Therefore, to conclude that $\mathcal{M}$ is an embedded manifold of $\mathbb{R}^n \times \mathbb{R}^n \cong \mathbb{C}^n$ with dimension $2n - k$, we need to show that f is a constant-rank function. Due to the smoothness of the function f, it is sufficient to show that $\mathbf{0}$ is a regular value of the function, i.e., the indefinite directional derivative of f at $\mathbf{x} \in \mathcal{M}$ is a surjective map. Given a direction $\xi_{\mathbf{x}} \in \mathbb{C}^n$, the directional derivative of f at $\mathbf{x}$ is given by

$$\mathsf{D}\ \mathsf{f}(\mathbf{x})[\xi_{\mathbf{x}}] = (\xi_{\mathbf{x}}^\star \mathbf{A}_1 \mathbf{x} + \mathbf{x}^\star \mathbf{A}_1 \xi_{\mathbf{x}}, \cdots, \xi_{\mathbf{x}}^\star \mathbf{A}_k \mathbf{x} + \mathbf{x}^\star \mathbf{A}_k \xi_{\mathbf{x}})^T.$$

Let $\mathbf{c} \in \mathbb{R}^k$ and define $\xi_{\mathbf{x}} \in \mathbb{C}^n$ by $\xi_{\mathbf{x}} = \frac{1}{2} \sum_{i=1}^{k} \frac{c_i}{b_i} \mathbf{A}_i \mathbf{x}$. The j-th entry in the vector $\mathsf{D}\ \mathsf{f}(\mathbf{x})[\xi_{\mathbf{x}}]$ is given by

$$\xi_{\mathbf{x}}^\star \mathbf{A}_j \mathbf{x} + \mathbf{x}^\star \mathbf{A}_j \xi_{\mathbf{x}} \overset{(a)}{=} \frac{c_j}{2b_j}(\mathbf{x}^\star \mathbf{A}_j \mathbf{x} + \mathbf{x}^\star \mathbf{A}_j \mathbf{x}) \overset{(b)}{=} c_j,$$

wherein the equality in (a) derives from the fact that the $\mathbf{A}_i$'s are orthogonal projection matrices and that the b_i's and c_i's are real numbers and the equality in (b) follows from the definition of $\mathbf{x} \in \mathcal{M}$. Therefore, we conclude that the indefinite directional derivative of f at $\mathbf{x} \in \mathcal{M}$ is a surjective map and that the tangent space is given by its kernel, i.e.,

$$\mathcal{T}_{\mathbf{x}}\mathcal{M} = \left\{ \xi_{\mathbf{x}} \in \mathbb{C}^n \mid \xi_{\mathbf{x}}^\star \mathbf{A}_i \mathbf{x} + \mathbf{x}^\star \mathbf{A}_i \xi_{\mathbf{x}} = 0,\ 1 \leq i \leq k \right\}.$$

■

Tangent spaces play an important role in Riemannian optimization in the same fashion that derivatives of smooth functions play a crucial role in numerical optimization.

The tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ allows to approximate the manifold in the neighborhood of $\mathbf{x} \in \mathcal{M}$ by a Euclidean space and to locally transform the constrained optimization into an unconstrained one. The *real* Riemannian metric g whose restriction to $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ is defined by,

$$\langle \xi_{\mathbf{x}}, \eta_{\mathbf{x}} \rangle_{\mathbf{x}} = \mathrm{Re}(\xi_{\mathbf{x}}^{\star}\eta_{\mathbf{x}}) = \frac{1}{2}(\xi_{\mathbf{x}}^{\star}\eta_{\mathbf{x}} + \eta_{\mathbf{x}}^{\star}\xi_{\mathbf{x}}), \tag{5.6}$$

turns $(\mathcal{M}, g)$ into a real smooth Riemannian manifold.

The normal space is defined as the orthogonal complement of the tangent space with respect to the Riemannian metric. In other words, the normal space at $\mathbf{x} \in \mathcal{M}$, denoted by $\mathcal{N}_{\mathbf{x}}\mathcal{M}$, is defined as the set of $\eta_{\mathbf{x}} \in \mathbb{C}^n$ such that $\langle \xi_{\mathbf{x}}, \eta_{\mathbf{x}} \rangle_{\mathbf{x}} = 0$ for all $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$. For the fixed norms manifold, the normal space has the following form,

$$\mathcal{N}_{\mathbf{x}}\mathcal{M} = \left\{ \eta_{\mathbf{x}} \in \mathbb{C}^n \mid \eta_{\mathbf{x}} = \sum_{i=1}^{k} \alpha_i \mathbf{A}_i \mathbf{x}, \ \{\alpha_i\}_{i=1}^{k} \in \mathbb{R} \right\}. \tag{5.7}$$

Indeed, it is easy to see that for any tangent vector $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$, and any normal vector $\eta_{\mathbf{x}} \in \mathcal{N}_{\mathbf{x}}\mathcal{M}$, the inner product gives

$$\langle \xi_{\mathbf{x}}, \eta_{\mathbf{x}} \rangle_{\mathbf{x}} \overset{(a)}{=} \frac{1}{2} \sum_{i=1}^{k} \alpha_i \left(\xi_{\mathbf{x}}^{\star} \mathbf{A}_i \mathbf{x} + \mathbf{x}^{\star} \mathbf{A}_i \xi_{\mathbf{x}} \right) \overset{(b)}{=} 0,$$

wherein the equality (a) is obtained from the fact that the $\mathbf{A}_i$'s are Hermitian and the α_i's are real numbers and (b) follows directly from the definition of the tangent space in (5.5). This shows that the normal space is included in the set identified in (5.7). Furthermore, we have the direct sum $\mathcal{T}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{N}_{\mathbf{x}}\mathcal{M} = \mathbb{C}^n$ with $\mathsf{Dim}\ (\mathcal{T}_{\mathbf{x}}\mathcal{M}) = 2n - k$, then $\mathsf{Dim}\ (\mathcal{N}_{\mathbf{x}}\mathcal{M}) = k$ which conclude the expression of the normal space in (5.7).

Regarding the fixed norms manifold as an embedded manifold in the Euclidean space $\mathbb{C}^n$ allows to exploit its linear structure to derive the required Riemannian operators to design optimization algorithms. The orthogonal projection from the linear embedding space to the linear tangent space plays an important role in deriving the Riemannian gradient, covariant derivative, etc. The projection for the fixed norms manifold is given in the following proposition.

Proposition 5.1 *Let* $\mathbf{y} \in \mathbb{C}^n$ *be an arbitrary complex vector, the projections of* $\mathbf{y}$ *onto the tangent space* $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ *and the normal space* $\mathcal{N}_{\mathbf{x}}\mathcal{M}$*, denoted by* $\Pi_{\mathbf{x}}(\mathbf{y})$ *and*

$\Pi_{\mathbf{x}}^{\perp}(\mathbf{y})$, *respectively, are given by,*

$$\Pi_{\mathbf{x}}(\mathbf{y}) = \mathbf{y} - \sum_{i=1}^{k} \frac{1}{2b_i}(\mathbf{y}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\mathbf{y})\mathbf{A}_i\mathbf{x}, \tag{5.8}$$

$$\Pi_{\mathbf{x}}^{\perp}(\mathbf{y}) = \sum_{i=1}^{k} \frac{1}{2b_i}(\mathbf{y}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\mathbf{y})\mathbf{A}_i\mathbf{x}. \tag{5.9}$$

Proof: Using the fact that the embedding space $\mathbb{C}^n$ is an Euclidean space satisfying $\mathcal{T}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{N}_{\mathbf{x}}\mathcal{M} = \mathbb{C}^n$, any arbitrary $\mathbf{y} \in \mathbb{C}^n$ can be decomposed as $\mathbf{y} = \Pi_{\mathbf{x}}(\mathbf{y}) + \Pi_{\mathbf{x}}^{\perp}(\mathbf{y})$ with the orthogonality condition $\langle \Pi_{\mathbf{x}}(\mathbf{y}), \Pi_{\mathbf{x}}^{\perp}(\mathbf{y}) \rangle_{\mathbf{x}} = 0$.

Recall that the tangent vector $\Pi_{\mathbf{x}}(\mathbf{y})$ satisfies $\Pi_{\mathbf{x}}^{\star}(\mathbf{y})\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\Pi_{\mathbf{x}}(\mathbf{y}) = 0,\ 1 \leq i \leq k$ and that the normal vector $\Pi_{\mathbf{x}}^{\perp}(\mathbf{y})$ is parameterized with $\Pi_{\mathbf{x}}^{\perp}(\mathbf{y}) = \sum_{i=1}^{k} \alpha_i \mathbf{A}_i\mathbf{x}$. Combining the decomposition of $\mathbf{y}$ with the parameterization of the tangent and normal space and the fact that $\mathbf{x} \in \mathcal{M}$ gives the expression of the reals $\alpha_i = \frac{1}{2b_i}(\mathbf{y}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\mathbf{y})$. Therefore, we obtain the projection on the normal space provided in (5.9) and thus the projection on the tangent space in (5.8). ■

Geodesics, Exponential, and Logarithmic Map

The exponential map is a function from a subset of a tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ to the manifold $\mathcal{M}$ that associates to each tangent direction $\xi_{\mathbf{x}}$ in the neighborhood of $\mathbf{0}_{\mathbf{x}}$ a geodesic curve $\gamma : \mathbb{R} \to \mathcal{M}$ going through $\mathbf{x} \in \mathcal{M}$ in the direction $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$, i.e., $\gamma(0) = \mathbf{x}$ and $\dot{\gamma}(0) = \xi_{\mathbf{x}}$. If the domain of the Exponential map is the whole tangent space, then the manifold is said to be geodesically complete. Hence, before deriving the exponential map, and its inverse known as the logarithmic map, one needs to define geodesics.

Geodesics are curves in $\mathcal{M}$ that generalize the concept of a “straight" line in $\mathbb{R}^n$. The concept of straight lines is generalized to curved space by the notion of curves with zero acceleration. While the velocity $\dot{\gamma}(t)$ of a curve $\gamma(t)$ can be computed easily through a usual derivative by exploiting the Euclidean structure of the embedding space, the notion of acceleration is more delicate. Indeed, the Euclidean definition of acceleration requires the differential of the vector field $\dot{\gamma}(t) : \mathbb{R} \to \mathcal{T}_{\gamma(t)}\mathcal{M}$. However, points $\dot{\gamma}(t)$ and $\dot{\gamma}(t + \delta t)$ are in two different tangent spaces, i.e., $\mathcal{T}_{\gamma(t)}\mathcal{M}$ and $\mathcal{T}_{\gamma(t+\delta t)}\mathcal{M}$. The notion of a covariant derivative is introduced to obtain a correspondence between the tangent spaces and compute the acceleration of curves.

The covariant derivative of a vector field is obtained by the Riemannian connection

∇ that generalizes the notion of directional derivative to vector fields. In particular, given two vector fields $\xi_{\mathbf{x}}$ and $\eta_{\mathbf{x}}$, the covariant derivative of $\xi_{\mathbf{x}}$ in the direction $\eta_{\mathbf{x}}$ is denoted by $\nabla_{\eta_{\mathbf{x}}}\xi_{\mathbf{x}}$. While there are infinitely many affine Riemannian connections, there exists a unique one that is symmetric and compatible with the Riemannian metric. Such connection is called the Levi-Civita connection and can be computed using Koszul formula. However, since the fixed norms manifold is embedded in an Euclidean space with a metric that does not depend on the tangent space, the Levi-Civita connection is given by,

$$\nabla_{\eta_{\mathbf{x}}}\xi_{\mathbf{x}} = \Pi_{\mathbf{x}}\left(\mathsf{D}\left(\xi_{\mathbf{x}}\right)[\eta_{\mathbf{x}}]\right).$$

With the above definition, finding the geodesic $\gamma_{\mathbf{x},\xi_{\mathbf{x}}}(t)$ going through $\mathbf{x} \in \mathcal{M}$ in the direction $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ amounts to solving the differential equation $\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = 0$ with the initial conditions $\gamma(0) = \mathbf{x}$ and $\dot{\gamma}(0) = \xi_{\mathbf{x}}$. The closed-form expression of $\gamma(t)$ is given in the following lemma.

Lemma 5.1 *The geodesic curve $\gamma_{\mathbf{x},\xi_{\mathbf{x}}} : \mathbb{R} \longrightarrow \mathcal{M}$ going through $\mathbf{x} \in \mathcal{M}$ in the direction $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ is given by*

$$\begin{aligned}\gamma_{\mathbf{x},\xi_{\mathbf{x}}}(t) = \sum_{i=1}^{k} \cos\left(\sqrt{\frac{\xi_{\mathbf{x}}^{\star}\mathbf{A}_i\xi_{\mathbf{x}}}{b_i}}t\right)\mathbf{A}_i\mathbf{x} \\ + \sqrt{\frac{b_i}{\xi_{\mathbf{x}}^{\star}\mathbf{A}_i\xi_{\mathbf{x}}}}\sin\left(\sqrt{\frac{\xi_{\mathbf{x}}^{\star}\mathbf{A}_i\xi_{\mathbf{x}}}{b_i}}t\right)\mathbf{A}_i\xi_{\mathbf{x}}\,. \qquad (5.10)\end{aligned}$$

Proof: Expanding the geodesic equality, $\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = \Pi_{\gamma(t)}(\ddot{\gamma}(t)) = 0$, gives the following differential equation,

$$\ddot{\gamma}(t) = \sum_{i=1}^{k} \frac{1}{2b_i}(\ddot{\gamma}^{\star}(t)\mathbf{A}_i\gamma(t) + \gamma^{\star}(t)\mathbf{A}_i\ddot{\gamma}(t))\mathbf{A}_i\gamma(t)\,. \qquad (5.11)$$

Let $c_i = \sqrt{\frac{\xi_{\mathbf{x}}^{\star}\mathbf{A}_i\xi_{\mathbf{x}}}{b_i}}$, $1 \le i \le k$ and define the curve $\gamma(t)$ by

$$\gamma(t) = \sum_{i=1}^{k} \cos\left(c_i t\right)\mathbf{A}_i\mathbf{x} + \frac{1}{c_i}\sin\left(c_i t\right)\mathbf{A}_i\xi_{\mathbf{x}}$$

The curve $\gamma(t)$ satisfies the following properties:

(i) $\gamma(0) = \sum_{i=1}^{k} \mathbf{A}_i\mathbf{x} = \mathbf{x}$.

(ii) $\dot{\gamma}(0) = \sum_{i=1}^{k} \mathbf{A}_i\xi_{\mathbf{x}} = \xi_{\mathbf{x}}$.

(iii) $\ddot{\gamma}(t) = \sum_{i=1}^{k} -c_i^2 \cos{(c_i t)}\, \mathbf{A}_i\mathbf{x} - c_i \sin{(c_i t)}\, \mathbf{A}_i\xi_{\mathbf{x}}$.

Furthermore, notice that $\mathbf{A}_i\gamma(t) = \cos{(c_i t)}\, \mathbf{A}_i\mathbf{x} + \frac{1}{c_i}\sin{(c_i t)}\, \mathbf{A}_i\xi_{\mathbf{x}}$, for all $1 \leq i \leq k$. Therefore, the second derivative can be expressed as $\ddot{\gamma}(t) = \sum_{i=1}^{k} -c_i^2\mathbf{A}_i\gamma(t)$. Finally, using the above properties, one can show that $\frac{1}{2b_i}(\ddot{\gamma}^\star(t)\mathbf{A}_i\gamma(t) + \gamma^\star(t)\mathbf{A}_i\ddot{\gamma}(t)) = -c_i^2$. Therefore, $\gamma(t)$ is a solution to the differential equation in (5.11) which concludes that it is the geodesic curve. ■

Given the expression of the geodesics on the fixed norms manifold in (5.10), the exponential map is defined as $\mathsf{Exp}_{\mathbf{x}}(\xi_{\mathbf{x}}) = \sum_{i=1}^{k} \cos{(c_i)}\, \mathbf{A}_i\mathbf{x} + \frac{1}{c_i}\sin{(c_i)}\, \mathbf{A}_i\xi_{\mathbf{x}}$ with $c_i = \sqrt{\frac{\xi_{\mathbf{x}}^\star\mathbf{A}_i\xi_{\mathbf{x}}}{b_i}}$, $1 \leq i \leq k$. Furthermore, since the geodesic curve is defined for all initial velocities $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$, then the exponential map is defined for the whole tangent space which makes $\mathcal{M}$ geodesically complete.

Geodesic Distance and First-Order Retraction

For a smooth curve $\gamma : [a, b] \to \mathcal{M}$ on the manifold $\mathcal{M}$, the length of the curve γ is defined as $\mathsf{L}\,(\gamma) = \int_a^b ||\dot{\gamma}(t)||_{\gamma(t)}\delta t$. Let $\mathbf{x}$ and $\mathbf{y}$ be two points on the manifold $\mathcal{M}$. The distance between $\mathbf{x}$ and $\mathbf{y}$, denoted by $d(\mathbf{x}, \mathbf{y})$, is defined the infimum of the lengths of all curves $\gamma(t)$ on $\mathcal{M}$ going through $\mathbf{x}$, i.e., $\gamma(0) = \mathbf{x}$, and ending at $\mathbf{y}$, i.e., $\gamma(1) = \mathbf{y}$. The distance operator is well defined for any couple of points $(\mathbf{x}, \mathbf{y}) \in \mathcal{M}^2$ that can be joined by a curve on $\mathcal{M}$ in which case the manifold is said to be connected.

If there exists a unique geodesic between two points $\mathbf{x}$ and $\mathbf{y}$ in $\mathcal{M}$, one can define the distance $d(\mathbf{x}, \mathbf{y})$ using the Logarithmic map by $d(\mathbf{x}, \mathbf{y}) = ||\mathsf{Log}_{\mathbf{x}}(\mathbf{y})||_{\mathbf{x}} = ||\mathsf{Log}_{\mathbf{y}}(\mathbf{x})||_{\mathbf{y}}$. Therefore, the distance on the fixed norms manifold between $\mathbf{x}$ and $\mathbf{y}$ such that $\{\mathbf{A}_i\Pi_{\mathbf{x}}(\mathbf{y})\}_{i=1}^{k}$ is not identically null, is given by the following expression

$$\begin{aligned} d(\mathbf{x}, \mathbf{y}) &= \sqrt{\mathsf{Log}_{\mathbf{x}}^\star(\mathbf{y})\mathsf{Log}_{\mathbf{x}}(\mathbf{y})} \qquad (5.12) \\ &= \sqrt{\sum_{i=1}^{k} b_i \arccos^2\left(\frac{1}{2b_i}(\mathbf{y}^\star\mathbf{A}_i\mathbf{x} + \mathbf{x}^\star\mathbf{A}_i\mathbf{y})\right)} \end{aligned}$$

The choice of a computationally efficient retraction is a crucial step in designing highly efficient Riemannian optimization algorithms. This chapter re-uses Theorem 3.2 to design the following highly efficient first-order retraction on the fixed norms manifold.

$$\mathsf{R}_{\mathbf{x}}(\xi_{\mathbf{x}}) = \sum_{i=1}^{k} \sqrt{\frac{b_i}{b_i + \xi_{\mathbf{x}}^{\star} \mathbf{A}_i \xi_{\mathbf{x}}}} \mathbf{A}_i \left(\mathbf{x} + \xi_{\mathbf{x}} \right) \tag{5.13}$$

Corollary 5.1 *The mapping* $\mathsf{R}_{\mathbf{x}}$ *defined in* (5.13) *is a retraction.*

Proof: Let $\mathcal{N} = \mathbb{R}_{++}^{k}$ be the set of element-wise strictly positive vectors of length k and define $\mathcal{E}^{\star} = \left\{ \mathbf{y} \in \mathbb{C}^n \mid \mathbf{y}^{\star} \mathbf{A}_i \mathbf{y} \neq 0,\ 1 \leq i \leq k \right\}$. Clearly, $\mathcal{E}^{\star}$ is an open subset of $\mathbb{C}^n$. Now define $\phi : \mathcal{M} \times \mathcal{N} \rightarrow \mathcal{E}^{\star}$ such that

$$\phi(\mathbf{x}, \mathbf{c}) = \left(\sum_{i=1}^{k} \sqrt{\frac{b_i}{c_i}} \mathbf{A}_i \right)^{-1} \mathbf{x}.$$

To show that ϕ is well defined, we need to show that any linear combination of the $\mathbf{A}_i$'s with non-zero weights is always full-rank. In other words, the matrix $\sum_{i=1}^{k} \beta_i \mathbf{A}_i$ is full-rank for non-zero β_i's. Assume it is not the case for some $\{\beta_i\}_{i=1}^{k}$. Then, there exists a non-zero vector $\mathbf{u}$ such that $\sum_{i=1}^{k} \beta_i \mathbf{A}_i \mathbf{u} = \mathbf{0}$. By applying $\mathbf{A}_j$ to the previous equation and using the orthogonality of the $\mathbf{A}_i$'s and the fact that they are projection matrices, we obtain $\beta_j \mathbf{A}_j \mathbf{u} = \mathbf{0}$. Since $\beta_j \neq 0$, we conclude that $\mathbf{A}_j \mathbf{u} = \mathbf{0}$ for all $1 \leq j \leq k$ which leads to $\sum_{j=1}^{k} \mathbf{A}_j \mathbf{u} = \mathbf{0} = \mathbf{I}_n \mathbf{u}$ and contradicts the fact that $\mathbf{u} \neq \mathbf{0}$.

It is straightforward to see that ϕ is a smooth and differentiable function with $\phi(\mathbf{x}, \mathbf{b}) = \mathbf{x}$ for all $\mathbf{x} \in \mathcal{M}$. Furthermore, given that $\left(\sum_{i=1}^{k} \sqrt{\frac{b_i}{c_i}} \mathbf{A}_i \right)$ is an invertible matrix, ϕ is a bijection whose differentiable inverse $(\mathbf{x}, \mathbf{c}) = \phi^{-1}(\mathbf{y})$ is given by

$$\mathbf{c}_j = \mathbf{y}^{\star} \mathbf{A}_j \mathbf{y},\ 1 \leq j \leq k$$

$$\mathbf{x} = \left(\sum_{i=1}^{k} \sqrt{\frac{b_i}{c_i}} \mathbf{A}_i \right) \mathbf{y}$$

According to Theorem 3.2, the first component of $\phi^{-1}(\mathbf{x}+\xi_{\mathbf{x}})$ represents the retraction of the tangent vector $\xi_{\mathbf{x}}$. Finally, using the expression of the manifold and its tangent space, the first component of $\phi^{-1}(\mathbf{x} + \xi_{\mathbf{x}})$ reduces to the expression in (5.13). ■

5.3 The Quotient Fixed Norms Manifold Geometry

The previous section studies the geometry of the manifold as embedded in the Euclidean space $\mathbb{C}^n$ in order to solve the phase retrieval problem in (5.2). However, the optimization problem in (5.2) presents non-isolated solutions. Indeed, solutions are invariant by left multiplication by a unit norm scalar. In other words, for any feasible solution $\mathbf{x} \in \mathcal{M}$, all points $e^{\mathrm{i}\theta}\mathbf{x}$ for $\theta \in [0, 2\pi)$ are also solutions with an identical cost function as $|\mathbf{a}^\star \mathbf{x}| = |e^{\mathrm{i}\theta}\mathbf{a}^\star \mathbf{x}|$. While the above-mentioned issue does not affect the convergence of first-order methods, it may not be benign to second-order algorithms such as Newton's or the TR methods. A popular approach to deal with the problem is to group all equivalent solutions $\mathbf{x}$ into an equivalence class $\overline{\mathbf{x}}$ and solve the optimization problem over these equivalence classes. This section accomplishes such goal by investigating the quotient geometry of the fixed norms manifold.

Equivalence Classes on Quotient Manifolds

Let $\sim$ be an equivalence relationship on the manifold $\mathcal{M}$ defined by $\mathbf{x}_1 \sim \mathbf{x}_2 \Leftrightarrow \exists\ \theta \in [0, 2\pi)$ s.t. $e^{\mathrm{i}\theta}\mathbf{x}_1 = \mathbf{x}_2$. The equivalence class of $\mathbf{x}$, i.e., all elements in relationship with $\mathbf{x}$, is denoted by $\overline{\mathbf{x}}$. Consider the set $\overline{\mathcal{M}} = \mathcal{M}/\sim$ defined as the set of all equivalence classes of points $\mathbf{x} \in \mathcal{M}$ by the relationship $\sim$. The mapping from points to their equivalence class $\pi : \mathcal{M} \to \overline{\mathcal{M}}$ is known as the canonical projection of points to their equivalence class, i.e., $\pi(\mathbf{x}) = \overline{\mathbf{x}}$.

Theorem 5.3 *The equivalence relation $\sim$ is regular for the fixed norms manifold $\mathcal{M}$ which provides the set $\overline{\mathcal{M}}$ with a manifold structure that makes it a quotient manifold of $\mathcal{M}$.*

Proof: Define the graph of an equivalence relationship by $\mathsf{graph}\ (\sim) = \{(\mathbf{x}_1, \mathbf{x}_2) \in \mathcal{M} \times \mathcal{M} \mid \mathbf{x}_1 \sim \mathbf{x}_2\}$. According to [23], a necessary and sufficient condition for $\overline{\mathcal{M}}$ to admit a quotient structure is that the equivalence relationship $\sim$ is regular on $\mathcal{M}$, i.e., it needs to satisfy the following properties:

1. $\mathsf{graph}\ (\sim)$ is an embedded submanifold of the product $\mathcal{M} \times \mathcal{M}$.

2. The projection $\pi_1 : \mathsf{graph}\ (\sim) \to \mathcal{M}$ given by $\pi_1(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1$ is a submersion.

3. $\mathsf{graph}\ (\sim)$ is closed.

We show that $\mathsf{graph}\ (\sim)$ is an embedded submanifold of $\mathcal{M} \times \mathcal{M}$ by applying the implicit function theorem in Theorem 5.2. Define $\Omega_i \subset \mathcal{M}$ such that for all $\mathbf{x} \in \Omega_i$, we have $\mathbf{x}_i \neq 0$. Clearly, we have $\cup_{i=1}^{k} \Omega_i = \mathcal{M}$. Define the mapping $\Omega_i \longrightarrow \mathcal{M}$ that associate to $\mathbf{x}$ the vector $\tilde{\mathbf{x}}_i = \mathbf{x}/\arg(\mathbf{x}_i)$. Define the function $f_i : \Omega_i \times \Omega_i \longrightarrow \mathbb{C}^n$ by $f_i(\mathbf{x}, \mathbf{y}) = \tilde{\mathbf{x}}_i - \tilde{\mathbf{y}}_i$. Clearly, we have $\mathsf{graph}\ (\sim) \cap \mathsf{dom}\ (f_i)$ is included in the level set $f_i^{-1}(\mathbf{0})$ as $\tilde{\mathbf{x}}_i = \tilde{\mathbf{y}}_i$ for any $(\mathbf{x}, \mathbf{y}) \in \mathsf{graph}\ (\sim)$. Furthermore, the equality $(\mathbf{x}, \mathbf{y}) \in f_i^{-1}(\mathbf{0})$ implies that $\tilde{\mathbf{x}}_i = \tilde{\mathbf{y}}_i$ which means that $\mathbf{x} = e^{\mathrm{i}\theta}\mathbf{y}$ for some $\theta \in [0, 2\pi)$, i.e., $(\mathbf{x}, \mathbf{y}) \in \mathsf{graph}\ (\sim) \cap \mathsf{dom}\ (f_i)$. Therefore, we obtain the equality $\mathsf{graph}\ (\sim) = \cup_{i=1}^{k} f_i^{-1}(\mathbf{0})$.

According to Lemma 5.2, we only need to show that indefinite directional derivative of f_i at $(\mathbf{x}, \mathbf{y}) \in \mathsf{graph}\ (\sim) \cap \mathsf{dom}\ (f_i)$, i.e., $\mathbf{x} = e^{\mathrm{i}\theta}\mathbf{y}$, is a surjective map. The directional derivative of $f_i(\mathbf{x}, \mathbf{y})$ in the direction ξ, η is given by

$$\begin{aligned}\mathsf{D}\ f_i(\mathbf{x}, \mathbf{y})[\xi, \eta] = {} & \frac{\xi}{\arg(\mathbf{x}_i)} - \frac{\mathrm{Re}(\xi_i)\,\mathrm{Re}(\mathbf{x}_i) - \mathrm{Im}(\xi_i)\,\mathrm{Im}(\mathbf{x}_i)}{\mathrm{Re}(\mathbf{x}_i)^2 \arg(\mathbf{x}_i)^2} \\ & - \frac{\eta}{\arg(\mathbf{y}_i)} + \frac{\mathrm{Re}(\eta_i)\,\mathrm{Re}(\mathbf{y}_i) - \mathrm{Im}(\eta_i)\,\mathrm{Im}(\mathbf{y}_i)}{\mathrm{Re}(\mathbf{y}_i)^2 \arg(\mathbf{y}_i)^2}\end{aligned}$$

For any $\mathbf{z} \in \mathbb{C}^n$, there exists ξ, η defined by $\eta_j = -\arg(\mathbf{y}_i)\mathbf{z}_j$ for $j \neq i$ and 0 otherwise and $\xi_j = 0$ for $j \neq i$ and $\xi_i = \frac{\mathrm{Re}(\mathbf{x}_i) \arg(\mathbf{x}_i)^2\, \mathrm{Re}(\mathbf{z}_i) - \mathrm{Im}(\mathbf{x}_i)\, \mathrm{Im}(\mathbf{z}_i)}{\mathrm{Re}(\mathbf{x}_i) \arg(\mathbf{x}_i) - 1} + \mathrm{i} \arg(\mathbf{y}_i)\mathbf{z}_i$ such that $\mathsf{D}\ f_i(\mathbf{x}, \mathbf{y})[\xi, \eta] = \mathbf{z}$ which concludes that f_i is a submersion and hence $\mathsf{graph}\ (\sim)$ is an embedded submanifold.

The fact that the projection π_1 is a submersion can be demonstrated by showing that the differential $\delta\pi_1$ is a surjective map. This can be accomplished by deriving a curve $\gamma_{\mathbf{y}}(t) : \mathcal{I} \longrightarrow \mathsf{graph}\ (\sim)$ for all $\mathbf{y} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ and all $\mathbf{x} \in \mathcal{M}$ such that

$$\left.\frac{\delta\pi_1(\gamma_{\mathbf{y}}(t))}{\delta t}\right|_{t=0} = \mathbf{y}. \tag{5.14}$$

Let $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$ be an equivalence class and consider two representatives $\mathbf{x}, \mathbf{x}' \in \pi^{-1}(\overline{\mathbf{x}})$ with $\mathbf{x}' = e^{\mathrm{i}\theta}\mathbf{x}$ for some $\theta \in [0, 2\pi)$. Consider the tangent vector $\mathbf{y} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ and define the curve $\gamma_{\mathbf{y}}(t) = \Big(\mathbf{x} + t\mathbf{y}, e^{\mathrm{i}\theta}(\mathbf{x} + t\mathbf{y})\Big) = (\mathbf{z}, \mathbf{z}')$. Clearly, we have $\mathbf{z} \sim \mathbf{z}'$. Furthermore, for any $1 \leq i \leq k$, we have the following

$$\begin{aligned}\mathbf{z}\mathbf{A}_i\mathbf{z}' &= (\mathbf{x} + t\mathbf{y})^{\star}\mathbf{A}_i(\mathbf{x} + t\mathbf{y}) \\ &= \mathbf{x}^{\star}\mathbf{A}_i\mathbf{x} + t(\mathbf{y}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\mathbf{y}) + o(t^2) \\ &= b_i + o(t^2).\end{aligned}$$

Therefore, for small-enough t, we have $\gamma_{\mathbf{y}}(t) \in \mathsf{graph}\ (\sim)$ while satisfying property (5.14) which concludes that π_1 is a submersion on $\mathcal{M}$.

Finally, to show that $\mathsf{graph}\ (\sim)$ is a closed set, it is sufficient to notice that $\mathsf{graph}\ (\sim)$ can be expressed as a level set of a smooth function. Define the function $f : \mathcal{M} \times \mathcal{M} \longrightarrow \mathbb{R}^{n\times n}$ by $\mathsf{f}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1\mathbf{x}_1^\star - \mathbf{x}_2\mathbf{x}_2^\star$. Clearly, we have $\mathsf{graph}\ (\sim)$ is included in the level set $f^{-1}(\mathbf{0})$ as $\mathbf{x}_1\mathbf{x}_1^\star - \mathbf{x}_2\mathbf{x}_2^\star = \mathbf{0}$ for any $(\mathbf{x}_1, \mathbf{x}_2) \in \mathsf{graph}\ (\sim)$. Furthermore, the equality $(\mathbf{x}_1, \mathbf{x}_2) \in f^{-1}(\mathbf{0})$ implies that $\mathbf{x}_1\mathbf{x}_1^\star = \mathbf{x}_2\mathbf{x}_2^\star$ which means that $\mathbf{x}_1 = e^{\mathrm{i}\theta}\mathbf{x}_2$ for some $\theta \in [0, 2\pi)$, i.e., $(\mathbf{x}_1, \mathbf{x}_2) \in \mathsf{graph}\ (\sim)$. Therefore, we obtain the equality $\mathsf{graph}\ (\sim) = f^{-1}(\mathbf{0})$. Finally, we conclude that $\mathsf{graph}\ (\sim)$ is closed by the fact that it is the image of a closed set by a smooth function. ■

Let $\mathcal{O}^n = \{\mathbf{O} \in \mathbb{C}^{n\times n} | \mathbf{O}\mathbf{O}^\star = \mathbf{O}^\star\mathbf{O} = \mathbf{I}\}$ be the set of unitary $n \times n$ matrices. In the rest of the chapter, the set $\overline{\mathcal{M}} = \mathcal{M}/\!\sim = \mathcal{M}/\mathcal{O}_1$ is called the quotient fixed norms manifold. The rest of this section investigates the different geometric elements of the quotient manifold that are crucial to design optimization algorithms.

Remark 5.1 *While Theorem 5.3 demonstrates that $\overline{\mathcal{M}}$ has a quotient structure, it does not imply that $\overline{\mathcal{M}}$ inherits the Riemannian structure of $\mathcal{M}$. This property is shown later on by demonstrating that the Riemannian metric in (5.6) is compatible with the equivalence relationship $\sim$.*

Horizontal and Vertical Spaces

Consider an equivalence class $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$, the class can be represented by multiple predecessors $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$. Similarly, the tangent vector $\xi_{\overline{\mathbf{x}}} \in \mathcal{T}_{\overline{\mathbf{x}}}\overline{\mathcal{M}}$ can be represented by different tangent vectors $\xi_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M}$ satisfying $\mathsf{D}\ (\pi(\mathbf{x}))[\xi_{\mathbf{x}}] = \xi_{\overline{\mathbf{x}}}$ for $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$. In order to obtain a unique representation of the tangent vector $\xi_{\overline{\mathbf{x}}}$ for each predecessor $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$, i.e., a unique solution to $\mathsf{D}\ (\pi(\mathbf{x}))[\xi_{\mathbf{x}}] = \xi_{\overline{\mathbf{x}}}$, one needs to decompose the tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ into a vertical $\mathcal{V}_{\mathbf{x}}\mathcal{M}$ and horizontal $\mathcal{H}_{\mathbf{x}}\mathcal{M}$ orthogonal components such that the representation is unique on the horizontal space. Such decomposition is possible thanks to the fact that $\pi^{-1}(\overline{\mathbf{x}})$ represents an embedded submanifold of $\mathcal{M}$ whose tangent space $\mathcal{T}_{\mathbf{x}}\pi^{-1}(\overline{\mathbf{x}})$ defines the vertical space. The horizontal space is the orthogonal, in the Riemannian metric sense, complement of the vertical space in $\mathcal{T}_{\mathbf{x}}\mathcal{M}$. In other words, a unique representation is obtained by decomposing the tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ such that

$$\begin{aligned} \mathcal{V}_{\mathbf{x}}\mathcal{M} &= \mathcal{T}_{\mathbf{x}}\pi^{-1}(\overline{\mathbf{x}}) \\ \mathcal{T}_{\mathbf{x}}\mathcal{M} &= \mathcal{V}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{H}_{\mathbf{x}}\mathcal{M} \end{aligned}$$

For the fixed norms manifold, the embedded submanifold of the predecessors of $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$ can be expressed as $\pi^{-1}(\overline{\mathbf{x}}) = \{\mathbf{y} \in \mathcal{M} \mid \mathbf{y} = e^{\mathrm{i}\theta}\mathbf{x}, \theta \in [0, 2\pi)\}$. As stated earlier, the vertical space is defined as the tangent space $\mathcal{T}_{\mathbf{x}}\pi^{-1}(\overline{\mathbf{x}})$. Consider the following curve $\gamma(t) : \mathcal{I} \longrightarrow \pi^{-1}(\overline{\mathbf{x}})$ such that $\gamma(t) = e^{\mathrm{i}\theta t}\mathbf{x}$. Clearly, we have $\gamma(t) \in \pi^{-1}(\overline{\mathbf{x}})$ for all $t \in \mathbb{R}$ and $\gamma(0) = \mathbf{x}$. The speed of the curve $\dot{\gamma}(0) = \mathrm{i}\theta e^{\mathrm{i}\theta t}\mathbf{x} \in \{\mathbf{y} \in \mathcal{M} \mid \mathbf{y} = \mathrm{i}\theta\mathbf{x}, \theta \in [0, 2\pi)\}$. Conversely, for $\mathbf{y} = \mathrm{i}\theta\mathbf{x}$, one can construct the curve $\gamma(t) = e^{\mathrm{i}\theta t}\mathbf{x}$ such that $\gamma(0) = \mathbf{x}$ and $\dot{\gamma}(0) = \mathbf{y}$. Therefore, we conclude that the vertical and horizontal spaces are given by

$$\mathcal{V}_{\mathbf{x}}\mathcal{M} = \{\xi_{\mathbf{x}} \in \mathbb{C}^n \mid \xi_{\mathbf{x}} = \mathrm{i}\theta\mathbf{x}, \text{ for } \theta \in [0, 2\pi)\} \tag{5.15}$$

$$\mathcal{H}_{\mathbf{x}}\mathcal{M} = \{\eta_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M} \mid \mathbf{x}^{\star}\eta_{\mathbf{x}} = \eta_{\mathbf{x}}^{\star}\mathbf{x}\} \tag{5.16}$$

Let $\xi_{\mathbf{x}} \in \mathcal{V}_{\mathbf{x}}\mathcal{M}$ and $\eta_{\mathbf{x}} \in \mathcal{H}_{\mathbf{x}}\mathcal{M}$ and define $\varphi_{\mathbf{x}} = \xi_{\mathbf{x}} + \eta_{\mathbf{x}}$. Notice that $\langle \xi_{\mathbf{x}}, \eta_{\mathbf{x}} \rangle_{\mathbf{x}} = 0$ and $\xi_{\mathbf{x}}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\xi_{\mathbf{x}} = -\mathrm{i}\theta b_i + \mathrm{i}\theta b_i = 0$. Therefore, we obtain $\mathcal{V}_{\mathbf{x}}\mathcal{M} \subset \mathcal{T}_{\mathbf{x}}\mathcal{M}$ and hence $\mathcal{V}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{H}_{\mathbf{x}}\mathcal{M} \subseteq \mathcal{T}_{\mathbf{x}}\mathcal{M}$. Finally, given that $\mathsf{Dim}\ (\mathcal{T}_{\mathbf{x}}\mathcal{M}) = 2n - k = 1 + (2n - k - 1) = \mathsf{Dim}\ (\mathcal{V}_{\mathbf{x}}\mathcal{M}) + \mathsf{Dim}\ (\mathcal{H}_{\mathbf{x}}\mathcal{M})$, we get the equality $\mathcal{V}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{H}_{\mathbf{x}}\mathcal{M} = \mathcal{T}_{\mathbf{x}}\mathcal{M}$ as anticipated.

Using the above decomposition of the tangent space, given $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$, the tangent vector $\xi_{\overline{\mathbf{x}}} \in \mathcal{T}_{\overline{\mathbf{x}}}\overline{\mathcal{M}}$ can be uniquely represented at $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$ by $\overline{\xi}_{\mathbf{x}} \in \mathcal{H}_{\mathbf{x}}\mathcal{M}$, called the horizontal lift of a tangent vector $\xi_{\overline{\mathbf{x}}}$ at $\mathbf{x}$ and satisfying $\mathsf{D}\ (\pi(\mathbf{x}))[\overline{\xi}_{\mathbf{x}}] = \xi_{\overline{\mathbf{x}}}$.

Lemma 5.2 *Consider the equivalence class* $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$ *and* $\mathbf{x}_1, \mathbf{x}_2 \in \pi^{-1}(\overline{\mathbf{x}})$ *such that* $\mathbf{x}_2 = e^{\mathrm{i}\theta}\mathbf{x}_1$ *for some* $\theta \in [0, 2\pi)$. *The horizontal lift* $\overline{\xi}_{\mathbf{x}_2}$ *of* $\xi_{\overline{\mathbf{x}}}$ *at* $\mathbf{x}_2$ *is related to the horizontal lift* $\overline{\xi}_{\mathbf{x}_1}$ *at* $\mathbf{x}_1$ *by*

$$\overline{\xi}_{\mathbf{x}_2} = \overline{\xi}_{e^{\mathrm{i}\theta}\mathbf{x}_1} = e^{\mathrm{i}\theta}\overline{\xi}_{\mathbf{x}_1} \tag{5.17}$$

Proof: Define the following set of equivalence classes

$$\mathcal{U}_{\mathbf{x}} = \{\overline{\mathbf{y}} \mid \mathbf{y}^{\star}\mathbf{A}_i\mathbf{y} = b_i, \mathbf{y}^{\star}\mathbf{x} = e^{\mathrm{i}\theta}, 1 \leq i \leq k, \theta \in [0, 2\pi)\}.$$

Let $\mathbf{y}_1, \mathbf{y}_2 \in \pi^{-1}(\overline{\mathbf{y}})$ such that $\mathbf{y}_2 = e^{\mathrm{i}\theta'}\mathbf{y}_1$. Assume that $\mathbf{y}_1^{\star}\mathbf{A}_i\mathbf{y}_1 = b_i$ and $\mathbf{y}_1^T\mathbf{x} = e^{\mathrm{i}\theta}$, then this implies that $\mathbf{y}_2^{\star}\mathbf{A}_i\mathbf{y}_2 = b_i$ and $\mathbf{y}_2^T\mathbf{x} = e^{\mathrm{i}\theta''}$ with $\theta'' = \theta + \theta' \in [0, 2\pi)$. In other words, the set $\mathcal{U}_{\mathbf{x}}$ is a well-defined set. Define the function $\sigma_{\mathbf{x}} : \mathcal{U}_{\mathbf{x}} \longrightarrow \mathbb{C}^n$ by $\sigma_{\mathbf{x}}(\overline{\mathbf{y}}) = \mathbf{y}\mathbf{y}^{\star}\mathbf{x}$. Using the same technique used for $\mathcal{U}_{\mathbf{x}}$, one can show that the expression of $\sigma_{\mathbf{x}}(\overline{\mathbf{y}})$ does not depend on the representative of the equivalence class $\overline{\mathbf{y}}$, i.e., the function $\sigma_{\mathbf{x}}$ is a well-defined function. Furthermore, it satisfies

$$\pi(\sigma_{\mathbf{x}}(\overline{\mathbf{y}})) = \pi(\mathbf{y}\mathbf{y}^{\star}\mathbf{x}) = \pi(e^{\mathrm{i}\theta}\mathbf{y}) = \overline{\mathbf{y}}$$

Therefore, $\sigma_{\mathbf{x}}$ is a right inverse of the natural projection π which implies that the operator $\mathsf{D}\,(\pi(\sigma_{\mathbf{x}}(\overline{\mathbf{y}}))) \circ \mathsf{D}\,(\sigma_{\mathbf{x}}(\overline{\mathbf{y}}))$ reduces to the identity map on $\mathcal{U}_{\mathbf{x}}$. Moreover, for any angle $\theta \in [0, 2\pi)$ notice the following

$$\sigma_{e^{\mathrm{i}\theta}\mathbf{x}}(\overline{\mathbf{y}}) = \mathbf{y}\mathbf{y}^{\star}e^{\mathrm{i}\theta}\mathbf{x} = e^{\mathrm{i}\theta}\mathbf{y}\mathbf{y}^{\star}\mathbf{x} = e^{\mathrm{i}\theta}\sigma_{\mathbf{x}}(\overline{\mathbf{y}}) \tag{5.18}$$

Define the set $\mathcal{S}_{\mathbf{x}} = \{\mathbf{z} \in \mathbb{C}^n \mid \mathbf{z} = \sigma_{\mathbf{x}}(\overline{\mathbf{y}}), \overline{\mathbf{y}} \in \mathcal{U}_{\mathbf{x}}\}$ as the image of $\mathcal{U}_{\mathbf{x}}$ by $\sigma_{\mathbf{x}}$. The set $\mathcal{S}_{\mathbf{x}}$ can be parameterized by

$$\mathcal{S}_{\mathbf{x}} = \{\mathbf{z} \in \mathbb{C}^n \mid \mathbf{z}^{\star}\mathbf{A}_i\mathbf{z} = b_i, \mathbf{z}^{\star}\mathbf{x} = \mathbf{x}^{\star}\mathbf{z}, 1 \leq i \leq k\}. \tag{5.19}$$

Indeed, let $\mathbf{z} = \sigma_{\mathbf{x}}(\overline{\mathbf{y}}) = \mathbf{y}\mathbf{y}^{\star}\mathbf{x} = e^{i\theta}\mathbf{y}$, then we have $\mathbf{z}^{\star}\mathbf{A}_i\mathbf{z} = e^{\mathrm{i}\theta}e^{-\mathrm{i}\theta}\mathbf{y}^{\star}\mathbf{A}_i\mathbf{y} = b_i$. Furthermore, the inner product $\mathbf{z}^{\star}\mathbf{x} = \mathbf{x}^{\star}\mathbf{y}\mathbf{y}^{\star}\mathbf{x} = \mathbf{x}^{\star}\mathbf{z}$. Finally, by noticing that both sets have dimension $2n - k - 1$, we conclude that $\mathcal{S}_{\mathbf{x}}$ has the expression provided in (5.19). The above parameterization allows to relate the horizontal space and the directional derivative of $\sigma_{\mathbf{x}}$ through the equation

$$\begin{aligned}\mathsf{D}\,\sigma_{\mathbf{x}}(\overline{\mathbf{y}}) &= \mathcal{T}_{\mathbf{x}}\mathcal{S}_{\mathbf{x}}\\ &= \{\eta_{\mathbf{x}} \in \mathbb{C}^n \mid \eta_{\mathbf{x}}^{\star}\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\eta_{\mathbf{x}} = \mathbf{0}, \eta_{\mathbf{x}}^{\star}\mathbf{x} = \mathbf{x}^{\star}\eta_{\mathbf{x}}, i \leq k\}\\ &= \{\eta_{\mathbf{x}} \in \mathcal{T}_{\mathbf{x}}\mathcal{M} \mid \eta_{\mathbf{x}}^{\star}\mathbf{x} = \mathbf{x}^{\star}\eta_{\mathbf{x}}\} = \mathcal{H}_{\mathbf{x}}\mathcal{M}\end{aligned} \tag{5.20}$$

Therefore, the horizontal lift $\overline{\xi}_{\mathbf{x}}$ can be expressed as $\overline{\xi}_{\mathbf{x}} = \mathsf{D}\,(\sigma_{\mathbf{x}})(\overline{\mathbf{y}})[\xi]$. Finally, using property (5.18), we conclude that the horizontal lift at $e^{\mathrm{i}\theta}\mathbf{x}$ for some $\theta \in [0, 2\pi)$ is related to the horizontal lift at $\mathbf{x}$ by

$$\overline{\xi}_{e^{\mathrm{i}\theta}\mathbf{x}} = e^{\mathrm{i}\theta}\overline{\xi}_{\mathbf{x}}$$

■

To show that the quotient manifold inherits the Riemannian structure of the embedded manifold, one needs to show that the Riemannian metric is compatible with the equivalence relationship. In other words, the Riemannian metric does not depend on the chosen representation of the class. To show that the property holds for the fixed norms manifold, consider an arbitrary equivalence class $\overline{\mathbf{x}}$, any two tangent vectors $\xi_{\overline{\mathbf{x}}}, \eta_{\overline{\mathbf{x}}} \in \mathcal{T}_{\overline{\mathbf{x}}}\overline{\mathcal{M}}$, and some predecessors $\mathbf{x}_1, \mathbf{x}_2 \in \pi^{-1}(\overline{\mathbf{x}})$, the horizontal lifts of $\xi_{\overline{\mathbf{x}}}$ and $\eta_{\overline{\mathbf{x}}}$ at $\mathbf{x}_1$ and $\mathbf{x}_2$ need to satisfy

$$\langle\overline{\xi}_{\mathbf{x}_1}, \overline{\eta}_{\mathbf{x}_1}\rangle_{\mathbf{x}_1} = \langle\overline{\xi}_{\mathbf{x}_2}, \overline{\eta}_{\mathbf{x}_2}\rangle_{\mathbf{x}_2}. \tag{5.21}$$

Given the expression of the horizontal lift in (5.17), it can clearly be seen that property (5.21) holds for any equivalence class $\overline{\mathbf{x}}$, any tangent vectors $\xi_{\overline{\mathbf{x}}}, \eta_{\overline{\mathbf{x}}} \in \mathcal{T}_{\overline{\mathbf{x}}}\overline{\mathcal{M}}$, and any predecessors $\mathbf{x}_1, \mathbf{x}_2 \in \pi^{-1}(\overline{\mathbf{x}})$. Therefore, the quotient manifold $\overline{\mathcal{M}}$ inherits the Riemannian structure of $\mathcal{M}$.

Levi-Civita Connection, Geodesics, and Exponential Map

This section derives the required geometric operator for the quotient fixed norms manifold so as to derive optimization algorithms over the manifold. In particular, this part first derives the expression of the Riemannian Levi-Civita connection. Afterward, the expression of the covariant derivative is utilized to solve the geodesic equation and provide an expression for the Exponential map.

Similar to the embedded fixed norms manifold, the covariant derivative $\nabla_{\eta_{\mathbf{x}}}\xi_{\mathbf{x}}$ of a vector field $\xi_{\mathbf{x}}$ in the direction $\eta_{\mathbf{x}}$ can be obtained from the Riemannian Levi-Citiva connection on the embedding Euclidean space which is given by $(\mathsf{D}\ (\xi_{\mathbf{x}})[\eta_{\mathbf{x}}]$ for the fixed norms manifold. However, instead of projecting the connection onto the whole tangent space, quotient manifold require the projection of the embedding connection onto the horizontal space. In other words, the Levi-Citiva connection on the quotient fixed norms manifold is

$$\nabla_{\eta_{\mathbf{x}}}\xi_{\mathbf{x}} = \Pi_{\mathbf{x}}^{\mathcal{H}}\left(\mathsf{D}\ (\xi_{\mathbf{x}})[\eta_{\mathbf{x}}]\right),$$

wherein $\Pi_{\mathbf{x}}^{\mathcal{H}}$ is the orthogonal projection onto the horizontal space at $\mathbf{x}$ whose expression is given in the following lemma.

Lemma 5.3 *The projection from the ambient space to the horizontal space is*

$$\Pi_{\mathbf{x}}^{\mathcal{H}}(\mathbf{y}) = \Pi_{\mathbf{x}}(\mathbf{y}) - \frac{\mathbf{x}^{\star}\mathbf{y} - \mathbf{y}^{\star}\mathbf{x}}{2\mathbf{x}^{\star}\mathbf{x}}\mathbf{x} \tag{5.22}$$

where $\Pi_{\mathbf{x}}(\mathbf{y})$ *is the orthogonal projection onto the tangent space* $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ *as defined in (5.8).*

Proof: This lemma is demonstrated through a direct computation of the orthogonal projection on the horizontal space. First, notice that the projection onto the horizontal space can be obtained as a projection onto the tangent space followed by a projection onto the horizontal space. For an ambient vector $\mathbf{y} \in \mathbb{C}^n$, the projection on the tangent space $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ for some $\mathbf{x} \in \mathcal{M}$ is given by $\Pi_{\mathbf{x}}(\mathbf{y})$ defined in (5.8). Now, the tangent vector $\Pi_{\mathbf{x}}(\mathbf{y})$ can be decomposed into a vertical and horizontal component using the fact that $\mathcal{T}_{\mathbf{x}}\mathcal{M} = \mathcal{V}_{\mathbf{x}}\mathcal{M} \oplus \mathcal{H}_{\mathbf{x}}\mathcal{M}$. Given the expression of the horizontal space in (5.16), we have $\xi_{\mathbf{x}} \in \mathcal{V}_{\mathbf{x}}\mathcal{M}$ implies that $\xi_{\mathbf{x}} = \mathrm{i}\theta\mathbf{x}$. Similarly, $\eta_{\mathbf{x}} \in \mathcal{H}_{\mathbf{x}}\mathcal{M}$ implies that $\eta_{\mathbf{x}}^{\star}\mathbf{x} = \mathbf{x}^{\star}\eta_{\mathbf{x}}$. Therefore, the projection $\Pi_{\mathbf{x}}(\mathbf{y})$ can be decomposed as $\Pi_{\mathbf{x}}(\mathbf{y}) = \xi_{\mathbf{x}} + \eta_{\mathbf{x}}$ with $\xi_{\mathbf{x}} = \mathrm{i}\theta\mathbf{x} = \dfrac{\mathbf{x}^{\star}\Pi_{\mathbf{x}}(\mathbf{y}) - \Pi_{\mathbf{x}}^{\star}(\mathbf{y})\mathbf{x}}{2\mathbf{x}^{\star}\mathbf{x}}\mathbf{x}$. With the above decomposition, the projection from the ambient space to the horizontal space is given by

$$\Pi^{\mathcal{H}}_{\mathbf{x}}(\mathbf{y}) = \Pi_{\mathbf{x}}(\mathbf{y}) - \frac{\mathbf{x}^\star\Pi_{\mathbf{x}}(\mathbf{y}) - \Pi^\star_{\mathbf{x}}(\mathbf{y})\mathbf{x}}{2\mathbf{x}^\star\mathbf{x}}\mathbf{x}.$$

Finally, we obtain the expression for the orthogonal projection onto the horizontal space in Lemma 5.3 by noticing the following equalities

$$\begin{aligned} \mathbf{x}^\star\Pi_{\mathbf{x}}(\mathbf{y}) &= \mathbf{x}^\star\mathbf{y} - \frac{1}{2}(\mathbf{y}^\star\mathbf{A}_i\mathbf{x} + \mathbf{x}^\star\mathbf{A}_i\mathbf{y}) \\ \Pi^\star_{\mathbf{x}}(\mathbf{y})\mathbf{x} &= \mathbf{y}^\star\mathbf{x} - \frac{1}{2}(\mathbf{y}^\star\mathbf{A}_i\mathbf{x} + \mathbf{x}^\star\mathbf{A}_i\mathbf{y}) \end{aligned}$$

■

The expressions of the Exponential map, Logarithmic map, and geodesic distance for the quotient fixed norms manifold are given in the following theorem.

Theorem 5.4 *Similar to the embedded fixed norms manifold, the expressions of the Exponential map, Logarithmic map, and geodesic distance for the quotient fixed norms manifold are*

$$\begin{aligned} \mathrm{Exp}_{\overline{\mathbf{x}}}(\xi_{\overline{\mathbf{x}}}) &= \sum_{i=1}^{k} \cos(c_i)\,\mathbf{A}_i\mathbf{x} + \frac{1}{c_i}\sin(c_i)\,\mathbf{A}_i\overline{\xi}_{\mathbf{x}} \\ \mathrm{Log}_{\overline{\mathbf{x}}}(\overline{\mathbf{y}}) &= \sum_{i=1}^{k} \frac{\arccos\left(\frac{1}{2b_i}(\mathbf{y}^\star\mathbf{A}_i\mathbf{x} + \mathbf{x}^\star\mathbf{A}_i\mathbf{y})\right)}{\sqrt{\frac{\Pi^\star_{\mathbf{x}}(\mathbf{y})\mathbf{A}_i\Pi_{\mathbf{x}}(\mathbf{y})}{b_i}}}\mathbf{A}_i\Pi_{\mathbf{x}}(\mathbf{y}) \\ d(\overline{\mathbf{x}}, \overline{\mathbf{y}}) &= \sqrt{\sum_{i=1}^{k} b_i \arccos^2\left(\frac{1}{2b_i}(\mathbf{y}^\star\mathbf{A}_i\mathbf{x} + \mathbf{x}^\star\mathbf{A}_i\mathbf{y})\right)}, \end{aligned}$$

with $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$, $\mathbf{y} \in \pi^{-1}(\overline{\mathbf{y}})$, $c_i = \sqrt{\frac{\xi^\star_{\mathbf{x}}\mathbf{A}_i\xi_{\mathbf{x}}}{b_i}}$, $1 \le i \le k$, *and* $\overline{\xi}_{\mathbf{x}} = \mathrm{Log}_{\overline{\mathbf{x}}}(\overline{\mathbf{y}})$.

Proof: Let $\overline{\mathbf{x}} \in \overline{\mathcal{M}}$ be an equivalence class and consider the tangent vector $\xi_{\overline{\mathbf{x}}} \in \mathcal{T}_{\overline{\mathbf{x}}}\overline{\mathcal{M}}$. The geodesic $\gamma_{\overline{\mathbf{x}},\xi_{\overline{\mathbf{x}}}}(t)$ going through $\overline{\mathbf{x}}$ in the direction $\xi_{\overline{\mathbf{x}}}$ is the solution to the differential equation $\nabla_{\dot\gamma(t)}\dot\gamma(t) = \Pi^{\mathcal{H}}_{\gamma(t)}(\ddot\gamma(t)) = 0$ with the initial conditions $\gamma(0) = \mathbf{x}$ and $\dot\gamma(0) = \overline{\xi}_{\mathbf{x}}$ with $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$ and $\overline{\xi}_{\mathbf{x}}$ being the horizontal lift of $\xi_{\overline{\mathbf{x}}}$ at $\mathbf{x}$.

Let $\gamma_{\mathbf{x},\xi_{\mathbf{x}}}(t)$ be the geodesic on the embedded fixed norms manifold going through $\mathbf{x}$ in the direction $\xi_{\mathbf{x}}$ whose expression in available in (5.10). The acceleration of the curve $\ddot{\gamma}_{\mathbf{x},\xi_{\mathbf{x}}}(t)$ satisfies

$$\begin{aligned}\ddot{\gamma}(t)^\star\gamma(t) &= \sum_{i=1}^{k} -c_i^2\cos^2(c_it)\,\mathbf{x}^\star\mathbf{A}_i\mathbf{x} - \sin^2(c_it)\,\xi_{\mathbf{x}}^\star\mathbf{A}_i\xi_{\mathbf{x}}\\ &= \sum_{i=1}^{k} -c_i^2\cos^2(c_it)\,b_i - \sin^2(c_it)\,\xi_{\mathbf{x}}^\star c_i^2 b_i\\ &= \sum_{i=1}^{k} -c_i^2 b_i = \gamma(t)^\star\ddot{\gamma}(t)\end{aligned}$$

In other words, the vector $\ddot{\gamma}_{\mathbf{x},\xi_{\mathbf{x}}}(t)$ is in the horizontal space $\mathcal{H}_{\gamma(t)}\mathcal{M}$, i.e., $\Pi^{\mathcal{H}}_{\gamma(t)}(\ddot{\gamma}(t)) = \Pi_{\gamma(t)}(\ddot{\gamma}(t))$. Therefore, the geodesic $\gamma_{\mathbf{x},\xi_{\mathbf{x}}}(t)$ satisfies the differential equation $\nabla_{\dot{\gamma}(t)}\dot{\gamma}(t) = \Pi^{\mathcal{H}}_{\gamma(t)}(\ddot{\gamma}(t)) = 0$, i.e., the expression of the geodesic on the quotient manifold is identical to the one on the embedded manifold.

Alternatively, one can one that the geodesic curve in (5.10) for the embedded fixed norms manifold is a solution to the differential equation $\Pi^{\mathcal{H}}_{\gamma(t)}(\ddot{\gamma}(t))$ for the geodesic on the quotient fixed norms manifold. Therefore, the expression of the Exponential map on the quotient manifold is identical to the one on the embedded manifold with the substitution of the equivalence class $\overline{\mathbf{x}}$ by a representative $\mathbf{x} \in \pi^{-1}(\overline{\mathbf{x}})$ and the tangent vector $\xi_{\overline{\mathbf{x}}}$ by its $\overline{\xi}_{\mathbf{x}}$ horizontal lift at $\mathbf{x}$. A similar remark allows to conclude that the expressions of the Logarithmic map and geodesic distance are identical to the previous ones, up to minor changes in notations as discussed above. ■

5.4 Phase Retrieval Algorithms and Numerical Results

Proposed Algorithms for Phase Retrieval

Piecing all the above ingredients together results in the Riemannian steepest-descent optimization algorithm on the embedded fixed norms manifold. For convenience, all the steps are summarized in Algorithm 5.1.

Simulation Results

To attest the performance of the proposed manifold in efficiently solving the Fourier phase retrieval problem, the convergence time and accuracy of the proposed gradient descent and conjugate gradient algorithms on the fixed norms manifold are compared to state-of-the-art unconstrained, e.g., TR, and constrained (interior-point and active set) optimization methods. The underlying signal $\mathbf{x}^\star$ is generated as a random complex Gaussian vector and all algorithms are initialized with $\mathbf{x}$ such that $\mathbb{E}||\mathbf{x} - \mathbf{x}^\star||_2^2 = 2n\sigma^2$.

Algorithm 5.1 Gradient descent on the fixed norms manifold.

```
Require: Manifold $\mathcal{M}$, loss function $\ell$, gradient Grad $\ell$.
Initialize $\mathbf{x} \in \mathcal{M}$.
while $|\text{Grad } \ell(\mathbf{x})||_{\mathbf{x}} \geq \epsilon$ do
    Compute search direction
    $$\xi_{\mathbf{x}} = -\text{Grad } \ell(\mathbf{x}) + \sum_{i=1}^{k} \frac{1}{2b_i}(\text{Grad } \ell^{\star}(\mathbf{x})\mathbf{A}_i\mathbf{x} + \mathbf{x}^{\star}\mathbf{A}_i\text{Grad } \ell(\mathbf{x}))\mathbf{A}_i\mathbf{x}$$
    Find Armijo step size $\alpha$ using Backtracking.
    Update $\mathbf{x} = \sum_{i=1}^{k} \sqrt{\frac{b_i}{b_i + \alpha^2 \xi_{\mathbf{x}}^{\star} \mathbf{A}_i \xi_{\mathbf{x}}}} \mathbf{A}_i \left(\mathbf{x} + \alpha \xi_{\mathbf{x}}\right)$.
end while
Output $\mathbf{x}$.
```

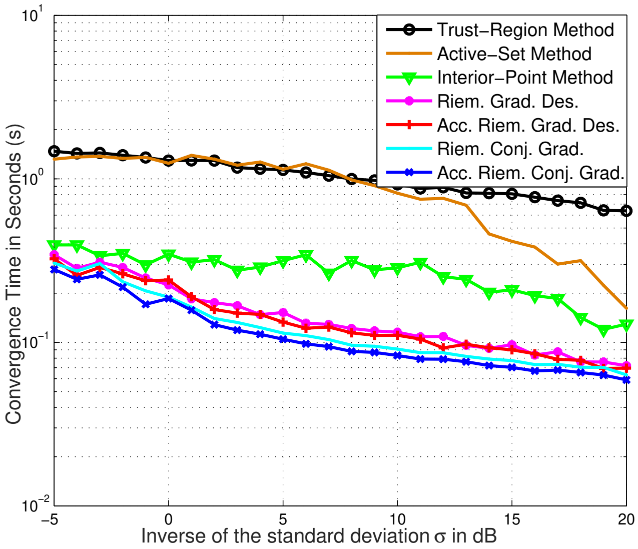


Figure 5.1: Running time to solve the Fourier phase retrieval problem.

Figure 5.1 plots the running time of the different algorithms in solving the Fourier phase retrieval problem. It can be seen from the figure that the proposed algorithms on the fixed norms manifold systematically run faster than all other tested algorithms with an average of 50 – 100 fold gain. Furthermore, it can be observed from Figure 5.2 that optimizing over the fixed norms manifold provides significantly higher accuracy, or equivalently a lower loss. For instance, for a σ equal to 5-dB, the achieved accuracy by the conjugate gradient on the fixed norms manifold is 7 order of magnitude higher than the best accuracy achieved by other tested algorithms. Hence, one can conclude that by properly exploiting the geometry of the problem,

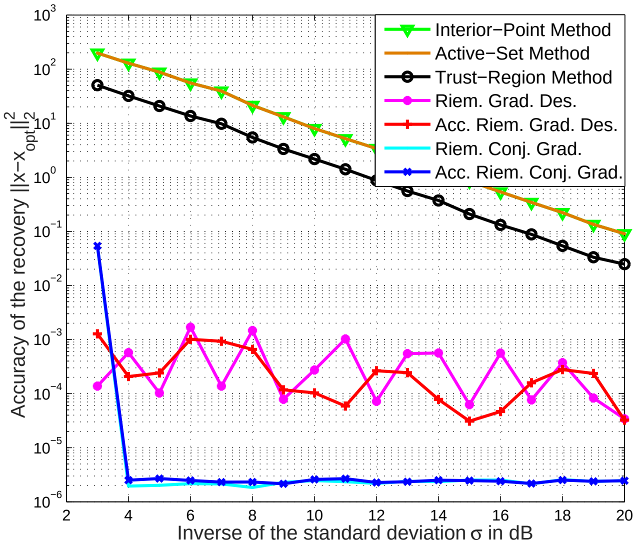


Figure 5.2: Accuracy in the reconstruction of the Fourier phase retrieval problem.

the proposed manifold outperforms traditional optimization-based methods both in accuracy and convergence speed.

Chapter 6

ACCURATE INDOOR AND OUTDOOR RIEMANNIAN LOCALIZATION

[1] A. Douik et al. “Precise 3-D GNSS Attitude Determination Based on Riemannian Manifold Optimization Algorithms”. In: *IEEE Transactions on Signal Processing* 68.1 (Dec. 2020), pp. 284–299. DOI: 10.1109/TSP.2019.2959226.

As its name indicates, this chapter uses Riemannian geometry to design efficient and accurate localization algorithms for indoor and outdoor applications. In particular, the thesis exploits the geometry of the receivers and transmitters to improve the accuracy of indoor and outdoor localization systems. Unlike previous works that consider each antenna separately, this chapter suggests incorporating the information about the geometry of the transmitters or receivers to perform a joint estimation under physical constraints on the placement of the antennas. The results of this section have been published in the research papers [130] and [52] for indoor and outdoor localization, respectively, and as such some of the text appears as it is in these publications.

6.1 Indoor Location Estimation Using Ultrasound Waves

Overview

Light-based localization systems, i.e., radio, infrared, and laser signals, suffer either from a low accuracy or a high hardware cost. Indeed, due to the top speed of light and without precise and costly synchronization, small timing errors result in significant localization errors [131]. As a result, localization systems based on Wi-Fi or Bluetooth technologies not only experience a low accuracy but also require pre-calibration [132]. Similarly, while radio-based approaches utilizing the Time of Flight (ToF) estimation do not require pre-calibration, these systems depend on an exact synchronization. Finally, laser and infrared-based localization devices are complicated and expensive to build and maintain[133]. This paper considers ultrasound-based localization methods [45] for their low-cost and high-accuracy, which is enabled by the relatively low speed of sound [134].

Besides the effects of the employed technology, the accuracy of indoor localization systems primarily depends on the optimization objective utilized in the design of those systems, e.g., see [135, 136] and references therein. For example, a simple approach consists of estimating the Received Signal Strength (RSS). While popular, the method suffers from poor localization accuracy due to multipath fading and temporal dynamics [137]. Alternatively, the Angle of Arrival (AoA) can be exploited to design high-accuracy systems for close-range location estimation. However, its performance significantly degrades as the distance between the transmitter and the receivers increases as a tiny error in the estimated angle results in a massive failure in the estimated position [138].

The previously mentioned ToF, defined as the time required for the signal to travel from the transmitter to the receiver, represents an attractive alternative approach for its simplicity. Still, a small perturbation in the estimated ToF can result in a significant deviation in the expected location, especially under a bad geometry [66]. To circumvent the aforementioned limitation, this manuscript uses multiple transmitters and considers exploiting their geometry in the estimation process. The resulting transmitter diversity not only significantly improves the accuracy of the estimated location but also provides the 3D orientation of the device.

In particular, this section considers a target with three transmitters that are placed on an equilateral triangle and utilizes a set of four receivers, known as beacons, to estimate the 3D location and orientation of the target accurately. The positions of the beacons are assumed to be identified correctly, and all distances between the transmitters and receivers are supposed to be estimated with a uniformly distributed error. These distances are fed to a classical nonlinear least-squares solver, such as the Gauss-Newton algorithm [139], to obtain an initial 3D location of the transmitters. Finally, the computed initialization is exploited by the designed non-convex Riemannian-based optimization algorithm to improve the location estimates of the three transmitters.

This section aims to design a novel and highly accurate spatial location estimation method using ultrasound waves. To that end, the transmitters' geometry is integrated into the location estimation process by formulating the problem as a non-convex optimization. Afterward, the set of feasible solutions is shown to admit a Riemannian manifold structure, which allows solving the underlying optimization problem rigorously. Hence, this thesis characterizes its geometry so as to design Riemannian optimization algorithms for the ultrasound spatial location estimation problem. The efficiency of the proposed method is attested through extensive simulations and comparisons with available algorithms in the literature. The numerical results suggest that the inclusion of the fixed equilateral triangle geometry of the transmitters as nonlinear constraints in the optimization problem significantly improves the quality of the location estimate. Furthermore, the proposed Riemannian-based method offers a clear complexity advantage as compared with popular generic non-convex approaches.

Related Work

A significant portion of indoor localization works that are available in the literature utilizes an array of receivers or transmitters to determine the location and orientation of a target. For example, in [140], the authors designed the Active Bat system, which estimates the position and orientation of an array of ultrasound transmitters based on the ToF estimation. The reference suggests performing a nonlinear regression combined with a least-squares solver to obtain the position of the target. Similarly, the Cricket system is introduced in [141]. The system consists of an array of ultrasound receivers that estimates the time of arrival (ToA) and angle of arrival (AoA) simultaneously. The time and angle of arrival are fed to nonlinear least-squares solver to obtain the location and orientation of the target. In [142], DOLPHIN has been introduced as an system to localize synchronized nodes in a typical indoor environment. DOLPHIN is similar to the Active Bat and Cricket systems except that it requires only few pre-configured reference nodes.

Combining the ToA and AoA as in [141] results in better performances, which explains their wide adoption in the literature. For example, Saad et al. [143] extend the method to mobile devices in a system that utilizes an array of three receivers. However, instead of relying on a nonlinear least-squares solution, the authors estimate the position of the target through a classical trilateration algorithm. Along the same lines, reference [144] considers an array of 8 receivers whose locations are estimated by a Taylor series trilateration method [145]. Finally, the mobile device's position and orientation are obtained based on averaging the AoA and location estimates for the 8 receivers. Such classical trilateration algorithm has been extended in [146] and [147] into a multilateration algorithm for a broadband localization system with multiple ultrasound transmitters and receivers.

The authors in [46] consider the problem of determining the position of a moving robot in a system comprising an equilateral ultrasound receiver array with a set of transmitters of known locations. The problem is solved by extending the Kalman

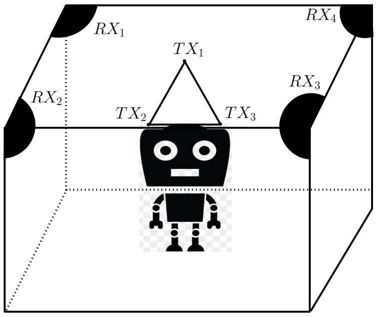


Figure 6.1: A localization system consisting of 3 transmitters and 4 receivers.

filter to incorporate a dynamic distance estimation method. The method of [46] is extended in [47] to design a time difference of arrival (TDoA)-based localization system. Thanks to the use of an interactive multiple model estimator, the authors report a precision of 2.5 to 3.5 cm.

To the best of the author's knowledge, all previously reported indoor localization systems do not impose the geometry of the receivers array as constraints when solving for the position of the target. Exploiting the fixed geometry of the transmitters or receivers array when formulating the optimization problem is expected to improve the localization estimation accuracy as long as the resulting non-convex problem can be solved efficiently. To that end, the rest of this section formulates the location determination problem as a non-convex program in which the constraints highlight the transmitters' geometry. Afterward, an efficient Riemannian-based optimization algorithm is designed by studying the geometry of the manifold derived from the set of all feasible solutions.

System Model and Problem Formulation

This section considers a localization system consisting of three ultrasound transmitters and four receivers, also known as beacons, and whose positions are perfectly known. The three transmitters form an equilateral triangle of size d as shown in Figure 6.1. The position of the target of interest is known with respect to the transmitters, e.g., it can be placed in the centroid of the triangle. Therefore, estimating the 3D locations of the three transmitters accurately provides an accurate 3D location and orientation of the target.

Let the 3D location of the i-th transmitter be $\mathbf{x}_i \in \mathbb{R}^3$. Likewise, let $\mathbf{b}_j \in \mathbb{R}^3$ denote the position of the j-th beacon. These positions are grouped in a matrix $\mathbf{A} \in \mathbb{R}^{4\times 3}$ such that the j-th row the matrix corresponds to the location of the j-th beacon, i.e.,

$\mathbf{a}_j^T = \mathbf{b}_j^T$ where the notation $\mathbf{z}^T$ refers to the transport of the vector $\mathbf{z}$.

The range r_{ij}, i.e., distance, from the i-th transmitter to the j-th beacon is estimated from the measurements on the received signal. It can readily be seen that the distance can be expressed using the vectors $\mathbf{x}_i$ and $\mathbf{b}_j$ as follows

$$r_{ij} = ||\mathbf{x}_i - \mathbf{b}_j||_2 = \sqrt{(\mathbf{x}_i - \mathbf{b}_j)^{\mathrm{T}}(\mathbf{x}_i - \mathbf{b}_j)}, \tag{6.1}$$

For ease of notations, the positions, distances, and measurements are collected in vectors are follows

$$\mathbf{x}_i^2 = ||\mathbf{x}_i||_2^2 \mathbf{1}_4 = \begin{pmatrix} ||\mathbf{x}_i||_2^2 \\ ||\mathbf{x}_i||_2^2 \\ ||\mathbf{x}_i||_2^2 \\ ||\mathbf{x}_i||_2^2 \end{pmatrix}, \mathbf{b}^2 = \begin{pmatrix} ||\mathbf{b}_1||_2^2 \\ ||\mathbf{b}_2||_2^2 \\ ||\mathbf{b}_3||_2^2 \\ ||\mathbf{b}_4||_2^2 \end{pmatrix}, \mathbf{r}_i^2 = \begin{pmatrix} r_{i1}^2 \\ r_{i2}^2 \\ r_{i3}^2 \\ r_{i4}^2 \end{pmatrix},$$

where $\mathbf{1}_4$ is the all ones vector of dimension 4. Define the transformed measurement vector $\mathbf{y}_i = \frac{1}{2}(\mathbf{b}^2 - \mathbf{r}_i^2)$. Using the expression of r_{ij}'s in (6.1), it can easily be concluded that

$$\mathbf{A}\mathbf{x}_i - \frac{1}{2}\mathbf{x}_i^2 = \mathbf{y}_i \in \mathbb{R}^{4\times 1}. \tag{6.2}$$

This section assumes that all distances between the transmitters and receivers, i.e., the r_{ij}'s, are supposed to be estimated with a uniformly distributed error. Therefore, a reasonable objective function is to consider the ℓ_2 loss between the measurement and the model (6.2). In other words, the paper consider the following objective function

$$\sum_{i=1}^{3} ||\mathbf{A}\mathbf{x}_i - \frac{1}{2}\mathbf{x}_i^2 - \mathbf{y}_i||_2^2.$$

The choice of the loss function to be used depend on the assumptions on the system model. While the manuscript focuses on the ℓ_2 loss, the results are more generic and can be applied to any smooth loss function as explained by the end of this section. To incorporate the fixed geometry of the transmitters, the spatial location estimation problem using ultrasound waves can be formulated as

$$\min_{\mathbf{x}_1,\mathbf{x}_2,\mathbf{x}_3\in\mathbb{R}^3} \sum_{i=1}^{3} ||\mathbf{A}\mathbf{x}_i - \frac{1}{2}\mathbf{x}_i^2 - \mathbf{y}_i||_2^2 \tag{6.3a}$$

$$\text{s.}t.\ (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = -d^2\cos\left(\frac{\pi}{3}\right) \tag{6.3b}$$

$$(\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = d^2\cos\left(\frac{\pi}{3}\right), \tag{6.3c}$$

wherein constraint (6.3b) insists on the length and angle of one side of the triangle while equation (6.3c) constraints the second side. It is worth noting that the constraint on the third side of the triangle is redundant as it can be obtained as a linear combination of constraints (6.3b) and (6.3c).

Despite the convexity of the objective function in (6.3a), the optimization problem is non-convex due to the quadratic nature of the constraints. Indeed, each feasible solution $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \mathbb{R}^3$ to (6.3) belongs to a set, named the *equilateral triangle manifold*, which is defined as follows:

$$\mathcal{M} = \left\{\{\mathbf{x}_i\}_{i=1}^3 \in \mathbb{R}^3 \,\middle|\, (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = -d^2\cos\left(\frac{\pi}{3}\right) \right.$$
$$\left. (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = d^2\cos\left(\frac{\pi}{3}\right)\right\}.$$

As stated earlier, the set $\mathcal{M}$ is non-convex. However, it forms a closed and bounded, i.e., a compact, matrix manifold embedded in the Euclidean space $\mathbb{R}^{3\times 3}$. Therefore, the study of the geometry of this newly introduced manifold allows to take advantage Riemannian optimization methods to efficiently solve the location estimation problem. Furthermore, instead of directly solving the optimization problem (6.3), this thesis suggests solving its generalization. In particular, let $\{\mathbf{x}_i\}_{i=1}^3$ be 3-dimensional vectors in $\mathbb{R}^3$ and consider a smooth function $f : \left(\mathbb{R}^3\right)^3 \longrightarrow \mathbb{R}$, that may or may not be convex, the rest of this section solves the optimization problem

$$\min_{\mathbf{x}_1,\mathbf{x}_2,\mathbf{x}_3\in\mathbb{R}^3} \mathrm{f}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \tag{6.4a}$$

$$\text{s.}t.\ (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \in \mathcal{M} \tag{6.4b}$$

The Equilateral Triangle Manifold Geometry

This part investigates and characterizes the first and second-order geometries of the equilateral triangle manifold so as to design optimization algorithms for the location estimation problem of interest. The first part shows that the set is indeed a manifold and computes its tangent space and orthogonal projection. The second and third parts compute the expression of the Riemannian gradient and Hessian and derive a computationally efficient retraction of the equilateral triangle manifold, respectively.

For conciseness and ease of notations, the variable $\mathbf{X} \in \mathbb{R}^{3\times 3}$ is used in the rest of this manuscript as a shorthand notation for the three vectors $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]$. Similarly, the tangent vector at $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]$ is denoted by $\xi_{\mathbf{X}} = [\xi_{\mathbf{x}_1}, \xi_{\mathbf{x}_2}, \xi_{\mathbf{x}_3}] \in \mathbb{R}^{3\times 3}$.

Recall that the equilateral triangle manifold is defined by

$$\begin{aligned}\mathcal{M} = \Big\{\mathbf{X} \in \mathbb{R}^{3\times 3} \Big| &(\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = -d^2 \cos\left(\frac{\pi}{3}\right) \\ &(\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) = d^2 \cos\left(\frac{\pi}{3}\right)\Big\}\end{aligned} \tag{6.5}$$

To show that the set defined in (6.5) is a well-defined manifold and to compute its tangent space, this section uses the implicit function theorem Theorem 5.2. Consider the function $g : \mathbb{R}^{3\times 3} \longrightarrow \mathbb{R}^2$ defined by

$$g(\mathbf{X}) = \begin{pmatrix} (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) + d^2 \cos\left(\frac{\pi}{3}\right) \\ (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) - d^2 \cos\left(\frac{\pi}{3}\right) \end{pmatrix}. \tag{6.6}$$

The above defined function only involves linear combinations and inner products of the vectors which makes it smooth as required by the implicit function theorem. In addition, the set of 2-dimensional vectors $\mathbb{R}^2$ is a linear space as mandated by Theorem 5.2. From the definition of the function and the set $\mathcal{M}$ is (6.5), it is clear that $\mathcal{M}$ is a level set of g as it can interpreted as the image of $\mathbf{0} \in \mathbb{R}^2$. Finally, showing that g is a constant-rank function can be accomplished by demonstrating that $\mathbf{0}$ is a regular value of g, i.e., the rank of each $\mathbf{X} \in g^{-1}(\mathbf{0}) = \mathcal{M}$ is equal to $\mathsf{Dim}\ (\mathbb{R}^2) = 2$ or equivalently that the indefinite directional derivative of g st any $\mathbf{X} \in \mathcal{M}$ is a surjective map.

Let $\mathbf{X} \in \mathcal{M}$ and consider an arbitrary direction $\xi_{\mathbf{X}} \in \mathbb{R}^{3\times 3}$, the directional derivative of g at $\mathbf{X}$ in the direction $\xi_{\mathbf{X}}$ is

$$\begin{aligned}&\mathsf{D}\ (g(\mathbf{X}))[\xi_{\mathbf{X}}] = \\ &\begin{pmatrix} (\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_2})^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) \\ (\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_3})^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) \end{pmatrix}.\end{aligned} \tag{6.7}$$

Let $\begin{pmatrix}\alpha\\\beta\end{pmatrix}$ be an arbitrary vector in $\mathbb{R}^2$. Finding $\xi_{\mathbf{X}} \in \mathbb{R}^{3\times3}$ such that $\mathsf{D}\,(g(\mathbf{X}))[\xi_{\mathbf{X}}] = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$ amounts to solving the following linear system of equations

$$\begin{pmatrix} \mathbf{x}_2 - \mathbf{x}_3 & \mathbf{x}_2 - \mathbf{x}_3 \\ \mathbf{x}_1 + \mathbf{x}_3 - 2\mathbf{x}_2 & \mathbf{x}_1 - \mathbf{x}_3 \\ \mathbf{x}_2 - \mathbf{x}_1 & 2\mathbf{x}_3 - \mathbf{x}_1 - \mathbf{x}_2 \end{pmatrix}^{\mathrm{T}} \begin{pmatrix}\xi_{\mathbf{x}_1}\\ \xi_{\mathbf{x}_2}\\ \xi_{\mathbf{x}_3}\end{pmatrix} = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

The above linear system has a fat matrix of dimension 2×3 and a full-rank 2. Indeed, assuming that the rank is equal to 1 gives the equality $\mathbf{x}_2 = \mathbf{x}_3$ which is impossible for any $\mathbf{X} \in \mathcal{M}$. Therefore, it holds true that the map $\mathsf{D}\,(g(\mathbf{X}))$ is surjective which concludes that the function g is a constant-rank function.

Finally, according to the results of Theorem 5.2, we conclude that the equilateral triangle set $\mathcal{M}$ defined in (6.5) is a well-defined manifold of dimension 7 embedded in the Euclidean space $\mathbb{R}^{3\times3}$. The tangent space is given by all directions $\xi_{\mathbf{X}} \in \mathbb{R}^{3\times3}$ such that nullify $\mathsf{D}\,(g(\mathbf{X}))[\xi_{\mathbf{X}}]$ which according to the expression given in (6.7) can be written as

$$\begin{aligned}\mathcal{T}_{\mathbf{X}}\mathcal{M} = \Big\{\xi_{\mathbf{X}} \in \mathbb{R}^{3\times3} \ \Big|& \\ &(\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_2})^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) = 0 \\ &(\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_3})^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) = 0\Big\}\end{aligned} \tag{6.8}$$

As stated earlier, this manuscript consider the induced Riemannian metric from the canonical inner product $\langle \mathbf{X}, \mathbf{X}\rangle = \mathrm{Tr}(\mathbf{X}^{\mathrm{T}}\mathbf{X})$ in $\mathbb{R}^{3\times3}$. In other words, the induced Riemannian metric on the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ is obtained from the natural embedding of $\mathcal{M}$ in $\mathbb{R}^{3\times3}$, i.e.,

$$\langle \xi_{\mathbf{X}}, \eta_{\mathbf{X}}\rangle_{\mathbf{X}} = \mathrm{Tr}(\xi_{\mathbf{X}}^{\mathrm{T}}\eta_{\mathbf{X}}) = \xi_{\mathbf{x}_1}^{\mathrm{T}}\eta_{\mathbf{x}_1} + \xi_{\mathbf{x}_2}^{\mathrm{T}}\eta_{\mathbf{x}_2} + \xi_{\mathbf{x}_3}^{\mathrm{T}}\eta_{\mathbf{x}_3}$$

The Riemannian gradient $\mathsf{grad}\ \mathsf{f}(\mathbf{X})$ can be expressed as the orthogonal projection of the Euclidean gradient from the embedding space $\mathbb{R}^{3\times3}$ to the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$. Let $\Pi_{\mathbf{X}} : \mathbb{R}^{3\times3} \longrightarrow \mathcal{T}_{\mathbf{X}}\mathcal{M}$ denote such orthogonal projection and note that the orthogonal complement $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}$ to the tangent space $\mathcal{T}_{\mathbf{X}}\mathcal{M}$ at $\mathbf{X} \in \mathcal{M}$ is given by the expression

$$\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M} = \Big\{\eta_{\mathbf{X}} \in \mathbb{R}^{3\times3} \ \Big|\eta_{\mathbf{X}} = \mathbf{X}\mathbf{U}_{\alpha}^{\beta}\Big\}$$

for some reals α and β and the matrix $\mathbf{U}_\alpha^\beta$ being defined as

$$\mathbf{U}_\alpha^\beta = \begin{pmatrix} 0 & \alpha+\beta & -\alpha-\beta \\ \alpha+\beta & -2\alpha & \alpha-\beta \\ -\alpha-\beta & \alpha-\beta & 2\beta \end{pmatrix}$$

Indeed, consider a tangent vector $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ and a normal vector $\eta_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}$, their inner product after expansion is given by

$$\begin{aligned} \langle \xi_{\mathbf{X}}, \eta_{\mathbf{X}} \rangle_{\mathbf{X}} &= \xi_{\mathbf{x}_1}^{\mathrm{T}} \eta_{\mathbf{x}_1} + \xi_{\mathbf{x}_2}^{\mathrm{T}} \eta_{\mathbf{x}_2} + \xi_{\mathbf{x}_3}^{\mathrm{T}} \eta_{\mathbf{x}_3} \\ &= \alpha \left[(\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_2})^{\mathrm{T}} (\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}} (\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) \right] \\ &\quad + \beta \left[(\xi_{\mathbf{x}_1} - \xi_{\mathbf{x}_3})^{\mathrm{T}} (\mathbf{x}_2 - \mathbf{x}_3) + (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}} (\xi_{\mathbf{x}_2} - \xi_{\mathbf{x}_3}) \right] \\ &= 0 \end{aligned}$$

Combine the above equality with the fact that $\mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}$ is an Euclidean space of dimension 2 allows to conclude that it represents the complement of the tangent space.

Let $\mathbf{Z} \in \mathbb{R}^{3\times 3}$ be a vector in the ambient space. Let $\mathbf{X} \in \mathcal{M}$ be a point on the manifold, the vector $\mathbf{Z}$ can be decomposed into a tangent part $\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z}_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ and an orthogonal part $\Pi_{\mathbf{X}}^{\perp}(\mathbf{Z}) = \mathbf{Z}_{\mathbf{X}}^{\perp} \in \mathcal{T}_{\mathbf{X}}^{\perp}\mathcal{M}$. From the previous analysis of the orthogonal complement of the tangent space, the orthogonal vector $\mathbf{Z}_{\mathbf{X}}^{\perp}$ is parametrized by two reals α and β such that $\mathbf{Z}_{\mathbf{X}}^{\perp} = \mathbf{X}\mathbf{U}_\alpha^\beta$. Finally, using the fact that the tangent vector $\mathbf{Z}_{\mathbf{X}}$ satisfies the equations in (6.8), we obtain that the reals α and β are the solution to

$$\begin{pmatrix} ||\mathbf{X}\mathbf{U}_1^0||_{\mathbf{X}}^2 & \langle \mathbf{X}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{X}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} & ||\mathbf{X}\mathbf{U}_0^1||_{\mathbf{X}}^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \mathbf{Z}, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{Z}, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{pmatrix}$$

The above linear system of equations admits a unique solution. Indeed, matrix $\mathbf{S}$ defined by $\mathbf{S} = \begin{pmatrix} ||\mathbf{X}\mathbf{U}_1^0||_{\mathbf{X}}^2 & \langle \mathbf{X}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{X}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} & ||\mathbf{X}\mathbf{U}_0^1||_{\mathbf{X}}^2 \end{pmatrix}$ is a positive semi-definite matrix with $\lambda_1^2 + \lambda_2^2 = ||\mathbf{S}||_F^2 \leq \mathrm{Tr}^2(\mathbf{S}) = (\lambda_1 + \lambda_2)^2$, i.e., $\lambda_1\lambda_2 \neq 0$. Therefore, the orthogonal projection onto the tangent space is given by

$$\Pi_{\mathbf{X}}(\mathbf{Z}) = \mathbf{Z} - \mathbf{X}\mathbf{U}_\alpha^\beta \tag{6.9}$$

with α and β defined as the solution to the linear system $\mathbf{S}\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \mathbf{Z}, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{Z}, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{pmatrix}$. Finally, applying the orthogonal projection $\Pi_{\mathbf{X}}$ to the Euclidean gradient $\mathsf{Grad}\ \mathsf{f}(\mathbf{X})$

results in the expression of the Riemannian gradient $\mathsf{grad}\ \mathsf{f}(\mathbf{X})$ as follows:

$$\mathsf{grad}\ \mathsf{f}(\mathbf{X}) = \mathsf{Grad}\ \mathsf{f}(\mathbf{X}) - \mathbf{X}\mathbf{U}_{\alpha}^{\beta} \tag{6.10}$$

with the reals α and β being the solution to the linear system

$$\begin{pmatrix} ||\mathbf{X}\mathbf{U}_1^0||_{\mathbf{X}}^2 & \langle \mathbf{X}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{X}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} & ||\mathbf{X}\mathbf{U}_0^1||_{\mathbf{X}}^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{pmatrix}$$

Given the expression of the Riemannian gradient in (6.10), the Riemannian Hessian can be computed as the orthogonal projection of the directional derivative of the Riemannian gradient. Let $\mathbf{X} \in \mathcal{M}$ be a vector on the manifold, $\xi_{\mathbf{X}} \in \mathcal{T}_{\mathbf{X}}\mathcal{M}$ a tangent vector, and $f : \mathcal{M} \longrightarrow \mathbb{R}$ a smooth function, the rest of this manuscript uses the short hand notation $\dot{f}(\mathbf{X})$ to denote the directional derivative $\mathsf{D}\ (\mathsf{f}(\mathbf{X}))\,[\xi_{\mathbf{X}}]$. Using the previously defined dot notation, the expression of the Riemannian Hessian is provided in the following corollary.

Corollary 6.1 *The Riemannian Hessian for the equilateral triangle manifold has the expression*

$$\mathsf{hess}\ \mathsf{f}(\mathbf{X})\,[\xi_{\mathbf{X}}] = \Pi_{\mathbf{X}}(\mathsf{Hess}\ \mathsf{f}(\mathbf{X})\,[\xi_{\mathbf{X}}] - \xi_{\mathbf{X}}\mathbf{U}_{\alpha}^{\beta} - \mathbf{X}\mathbf{U}_{\dot{\alpha}}^{\dot{\beta}}) \tag{6.11}$$

wherein the expression of the orthogonal projection $\Pi_{\mathbf{X}}$ *is given in (6.9),*

$$\mathbf{S} = \begin{pmatrix} ||\mathbf{X}\mathbf{U}_1^0||_{\mathbf{X}}^2 & \langle \mathbf{X}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{X}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} & ||\mathbf{X}\mathbf{U}_0^1||_{\mathbf{X}}^2 \end{pmatrix}$$

and α *and* β *are the solution to*

$$\mathbf{S}\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{pmatrix}$$

and their directional derivatives $\dot{\alpha}$ *and* $\dot{\beta}$ *are the solution to the system*

$$\mathbf{S}\begin{pmatrix} \dot{\alpha} \\ \dot{\beta} \end{pmatrix} = \begin{pmatrix} \langle \mathsf{Hess}\ \mathsf{f}(\mathbf{X})\,[\xi_{\mathbf{X}}], \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} + \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \xi_{\mathbf{X}}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathsf{Hess}\ \mathsf{f}(\mathbf{X})\,[\xi_{\mathbf{X}}], \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} + \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \xi_{\mathbf{X}}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{pmatrix} - \dot{\mathbf{S}}\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$$

and the directional derivative of $\mathbf{S}$ *in the direction* $\xi_{\mathbf{X}}$ *is*

$$\dot{\mathbf{S}} = \left(\begin{array}{c|c} 2\langle \xi_{\mathbf{X}}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} & \begin{array}{c} \langle \xi_{\mathbf{X}}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \\ +\langle \mathbf{X}\mathbf{U}_0^1, \xi_{\mathbf{X}}\mathbf{U}_1^0 \rangle_{\mathbf{X}} \end{array} \\ \hline \begin{array}{c} \langle \xi_{\mathbf{X}}\mathbf{U}_1^0, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \\ +\langle \mathbf{X}\mathbf{U}_1^0, \xi_{\mathbf{X}}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{array} & 2\langle \xi_{\mathbf{X}}\mathbf{U}_0^1, \mathbf{X}\mathbf{U}_0^1 \rangle_{\mathbf{X}} \end{array}\right)$$

Proof: The proof of this corollary is omitted herein as it follows from a direct computation of the orthogonal projection of the directional derivative of the Riemannian gradient. ■

Retraction on the Equilateral Triangle Manifold

Designing a computationally efficient retraction is a crucial step in deriving Riemannian optimization algorithms. While it is relatively easy to design functions that are local retraction around $\mathbf{0}_{\mathbf{X}}$, e.g., $\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \mathbf{X} + \xi_{\mathbf{X}}$, these retractions often result in Riemannian algorithms with poor performance. Indeed, the resulting iterative optimization algorithm would generate smaller and smaller optimization steps ultimately converging before reaching a critical point of the problem. Luckily, for manifolds defined with only equality constraints such as the equilateral triangle manifold of interest herein, Theorem 3.2 can be exploited to design efficient retractions.

To that end, define $\mathcal{E}^*$ as a subset of $\mathbb{R}^{3\times3}$ such that $\mathbf{Z} \in \mathcal{E}^*$ implies that $\mathbf{z}_2 \neq \pm\mathbf{z}_3$. It can easily be seen that $\mathcal{E}^*$ is an open subset of $\mathbb{R}^{3\times3}$. Furthermore, let $\mathcal{N} = \mathbb{R}^2_*$ be the set of 2-dimensional vectors $\begin{pmatrix}\alpha\\ \beta\end{pmatrix}$ such that $\alpha \neq 0$ and $\beta \neq 0$. Since $\dim(\mathcal{N}) = 2$, the property $\dim(\mathcal{M}) + \dim(\mathcal{N}) = \dim(\mathcal{E})$ is satisfied. Now define the function

$$\phi\left(\mathbf{X}, \begin{pmatrix}\alpha\\ \beta\end{pmatrix}\right) = \begin{pmatrix} \begin{pmatrix}\alpha\beta & 0 & 0\\ 0 & \alpha\beta & 0\\ \mathbf{0} & \mathbf{0} & \beta\end{pmatrix}\mathbf{x}_1 \\ \beta\mathbf{x}_2 \\ \beta\mathbf{x}_3 \end{pmatrix}$$

Note that for any $\mathbf{X} \in \mathcal{M}$, we have $\phi\left(\mathbf{X}, \begin{pmatrix}1\\ 1\end{pmatrix}\right) = \mathbf{X}$ as mandated by Theorem 3.2. In addition, the smoothness of the function ϕ directly derives from its definition as it involves only products. Now let $\mathbf{Z}$ be an arbitrary vector in $\mathcal{E}^*$. From the expression of ϕ is can easily be seen that the first term of the inverse $\pi_1(\phi^{-1})$ can be written as

$$\begin{pmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \mathbf{x}_3\end{pmatrix} = \lambda \begin{pmatrix} \begin{pmatrix}\gamma & 0 & 0\\ 0 & \gamma & 0\\ 0 & 0 & 1\end{pmatrix}\mathbf{z}_1 \\ \mathbf{z}_2 \\ \mathbf{z}_3 \end{pmatrix}$$

for some λ and γ functions of $\mathbf{Z}$. Therefore, the inverse of the first component is smooth. Consider the transformation $\mathbf{u}_1 = \begin{pmatrix} \gamma & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{z}_1$, $\mathbf{u}_2 = \mathbf{z}_2$, and $\mathbf{u}_3 = \mathbf{z}_3$. It is easy to see that there exists a unique γ such that the equality $(\mathbf{u}_1 - \mathbf{u}_2)^{\mathrm{T}}(\mathbf{u}_2 - \mathbf{u}_3) = -(\mathbf{u}_1 - \mathbf{u}_3)^{\mathrm{T}}(\mathbf{u}_2 - \mathbf{u}_3)$ is satisfied. The expression of γ is given by

$$\gamma = \frac{(\mathbf{z}_2 + \mathbf{z}_3)^{\mathrm{T}}(\mathbf{z}_2 - \mathbf{z}_3) - \mathbf{z}_1^T(\mathbf{z}_2 - \mathbf{z}_3)}{2\mathbf{z}_1^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} (\mathbf{z}_2 - \mathbf{z}_3)} - 1$$

Finally, the point $\mathbf{X} \in \mathcal{M}$ is obtained by scaling the vector $\mathbf{U}$ by the quantity $\lambda = \sqrt{\frac{d^2 \cos\left(\frac{\pi}{3}\right)}{(\mathbf{u}_1 - \mathbf{u}_3)^{\mathrm{T}}(\mathbf{u}_2 - \mathbf{u}_3)}}$, i.e., $\mathbf{X} = \lambda\mathbf{U}$ to obtain the manifold characterization

$$\begin{aligned} (\mathbf{x}_1 - \mathbf{x}_2)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) &= -d^2 \cos\left(\frac{\pi}{3}\right) \\ (\mathbf{x}_1 - \mathbf{x}_3)^{\mathrm{T}}(\mathbf{x}_2 - \mathbf{x}_3) &= d^2 \cos\left(\frac{\pi}{3}\right) \end{aligned}$$

Since the expressions of γ and λ are rational functions of the argument $\mathbf{Z}$ without any pole as $\mathbf{z}_2 \neq \pm\mathbf{z}_3$ and that $\mathbf{X}$ is obtained by a simple multiplication, it can be concluded that ϕ^{-1} is smooth which gives that ϕ is a diffeomorphism as requested by Theorem 3.2. Finally, combining all the results above and letting $\mathbf{Z} = \mathbf{X} + \xi_{\mathbf{X}}$, this section proposes the following retraction

$$\mathsf{R}_{\mathbf{X}}(\xi_{\mathbf{X}}) = \sqrt{\frac{d^2 \cos\left(\frac{\pi}{3}\right)}{\lambda}} \begin{pmatrix} \begin{pmatrix} \gamma & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \mathbf{z}_3 \end{pmatrix} \tag{6.12}$$

with

$$\lambda = \left(\begin{pmatrix} \gamma & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{z}_1 - \mathbf{z}_3 \right)^{\mathrm{T}} (\mathbf{z}_2 - \mathbf{z}_3)$$

$$\gamma = \frac{(\mathbf{z}_2 + \mathbf{z}_3)^{\mathrm{T}} (\mathbf{z}_2 - \mathbf{z}_3) - \mathbf{z}_1^T (\mathbf{z}_2 - \mathbf{z}_3)}{2\mathbf{z}_1^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} (\mathbf{z}_2 - \mathbf{z}_3)} - 1$$

Initialization and Proposed Algorithm

Incorporating the geometry of the receivers in the optimization problem turns the problem into a non-convex program. The performance of non-convex problems heavily relies on the quality of the initialization. An random initialization on the manifold can obtained by generating a random orthonormal matrix $\mathbf{o} \in \mathbb{R}^{3\times 3}$, i.e., $\mathbf{o}\mathbf{o}^{\mathrm{T}} = \mathbf{I}$, and initializing $\mathbf{X} = d\sqrt{\cos\left(\frac{\pi}{3}\right)}\mathbf{o}$. However, due to the non-convex nature of the optimization problem, better results can be obtained by using an improved initialization.

This section proposes finding an improved initialization by solving the location problem without constraints on the geometry of the transmitters. This can be accomplished using a classical non-linear least-squares solver, such as the Gauss-Newton algorithm [139]. Let $\tilde{\mathbf{X}}_0$ be the solution obtained without constraints on the geometry of the transmitters. Such solution does not necessarily belong to the equilateral triangle manifold. Therefore, the second step in deriving an improved initialization is to “project" the point $\tilde{\mathbf{X}}_0$ to the manifold. This is accomplished by solving the optimization problem

$$\mathbf{X}_0 = \arg\min_{\mathbf{X}\in\mathcal{M}} ||\mathbf{X} - \tilde{\mathbf{X}}_0||_2^2 \tag{6.13}$$

The optimization problem in (6.13) can be efficiently solved using the geometry derived previously. Indeed, as pointed out previously, the proposed framework allows to optimize any objective function over the equilateral triangle manifold, including the function $||\mathbf{X} - \tilde{\mathbf{X}}_0||_2^2$. Random initialization on the manifold, as described above, can be used to solve (6.13). The steps of the algorithm are omitted herein as they are provided and described in the next subsection.

Note that the initialization highly depend on the assumptions on the system and the considered loss function. In other words, while the proposed initialization in (6.13) performs well for the considered ℓ_2 loss in (6.3), it might not be optimal for different objective functions.

The algorithm starts by initialization $\mathbf{X} = \mathbf{X}_0 \in \mathcal{M}$. Afterward, the algorithm iterates on finding a search direction and updating the current position. The search direction is given by $\xi_{\mathbf{X}} = -\frac{\mathsf{grad}\ \mathsf{f}(\mathbf{X})}{||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}}}$ wherein the Riemannian gradient is computed according to (6.10). The step size t is chosen by backtracking so as it satisfies the following Wolfe conditions

1. $\mathsf{f}(\mathbf{X} + t\xi_{\mathbf{X}}) \leq \mathsf{f}(\mathbf{X}) + c_1 t \xi_{\mathbf{X}}^{\mathrm{T}} \mathsf{Grad}\ \mathsf{f}(\mathbf{X})$
2. $-\xi_{\mathbf{X}}^{\mathrm{T}} \mathsf{Grad}\ \mathsf{f}(\mathbf{X} + t\xi_{\mathbf{X}}) \leq -c_2 \xi_{\mathbf{X}}^{\mathrm{T}} \mathsf{Grad}\ \mathsf{f}(\mathbf{X})$,

for some constants $0 < c_1 < c_2 < 1$. The tangent vector $\xi_{\mathbf{X}}$ scaled with the step size t is retracted to the manifold using (6.12) to update the position $\mathbf{X}$. The process is repeated until convergence which can be attested by the norm of the Riemannian gradient. The steps of the proposed Riemannian steepest-descent algorithm are summarized in Algorithm 6.1.

It is worth noting that unlike other non-convex methods whose complexity is at least quadratic in the number of variables n, all the steps in our proposed Riemannian method are linear in the number of variables resulting in an overall algorithm with linear complexity. Furthermore, recall that Newton's method on the equilateral triangle manifold is obtained by choosing the tangent vector that solves $\mathsf{hess}\ \mathsf{f}(\mathbf{X})[\xi_{\mathbf{X}}] = -\mathsf{grad}\ \mathsf{f}(\mathbf{X})$. Since the previous step can be accomplished in n^2 operations, our proposed Riemannian Newton's method is quadratic in the number of variables which competes with the complexity of first-order generic non-convex solvers, e.g., interior-point method (IPM).

Simulation Setup and Numerical Results

This part presents the simulation results to evaluate the proposed algorithm against some benchmark algorithms in a noisy environment. We evaluate the performance of the proposed algorithm against some benchmark algorithms, namely the interior-point method (IPM) [148, 149, 150], the active set algorithm [151], and the sequential quadratic programming (SQP) algorithm [151]. Moreover, the paper illustrates

Algorithm 6.1 Riemannian steepest-descent on the equilateral triangle manifold.

```
Require: Length $d > 0$, initialization $\mathbf{X}_0$, a tolerance $\epsilon > 0$, and a smooth function $f$.
Initialize $\mathbf{X} = \mathbf{X}_0 \in \mathcal{M}$.
while $||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}} \neq \epsilon$ do
    Find $\alpha$ and $\beta$ by solving
        $$\begin{pmatrix} ||\mathbf{XU}_1^0||_{\mathbf{X}}^2 & \langle \mathbf{XU}_0^1, \mathbf{XU}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathbf{XU}_1^0, \mathbf{XU}_0^1 \rangle_{\mathbf{X}} & ||\mathbf{XU}_0^1||_{\mathbf{X}}^2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{XU}_1^0 \rangle_{\mathbf{X}} \\ \langle \mathsf{Grad}\ \mathsf{f}(\mathbf{X}), \mathbf{XU}_0^1 \rangle_{\mathbf{X}} \end{pmatrix}$$
    Compute the Riemannian gradient
        $$\mathsf{grad}\ \mathsf{f}(\mathbf{X}) = \mathsf{Grad}\ \mathsf{f}(\mathbf{X}) - \mathbf{X} \begin{pmatrix} 0 & \alpha + \beta & -\alpha - \beta \\ \alpha + \beta & -2\alpha & \alpha - \beta \\ -\alpha - \beta & \alpha - \beta & 2\beta \end{pmatrix}$$
    Set search direction $\xi_{\mathbf{X}} = -\frac{\mathsf{grad}\ \mathsf{f}(\mathbf{X})}{||\mathsf{grad}\ \mathsf{f}(\mathbf{X})||_{\mathbf{X}}}$
    Compute the step size $t$ using backtracking.
    Define $\mathbf{Z} = \mathbf{X} + t\xi_{\mathbf{X}}$ and compute
        $$\gamma = \frac{(\mathbf{z}_2 + \mathbf{z}_3)^{\mathrm{T}}(\mathbf{z}_2 - \mathbf{z}_3) - \mathbf{z}_1^T(\mathbf{z}_2 - \mathbf{z}_3)}{2\mathbf{z}_1^T \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} (\mathbf{z}_2 - \mathbf{z}_3)} - 1$$
    Define the equilateral triangle $\mathbf{U}$ by
        $$[\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3] = \left[ \begin{pmatrix} \gamma & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3 \right]$$
    Scale the sides of $\mathbf{U}$ to obtain a triangle in $\mathcal{M}$ by
        $$\mathbf{X} = \sqrt{\frac{d^2 \cos\left(\frac{\pi}{3}\right)}{(\mathbf{u}_1 - \mathbf{u}_3)^{\mathrm{T}}(\mathbf{u}_2 - \mathbf{u}_3)}}\,\mathbf{U}$$
end while
```

the improvement in the location estimation accuracy as compared to the commonly used trilateration algorithm which utilizes Gauss-Newton (GN) method [145].

Besides the previously mentioned steepest-descent and Newton's algorithms on manifolds, this section implements the Riemannian version of the trust-region [152] and Adaptive Regularization with Cubics (ARC) methods [153]. These methods can readily be implemented using the derived geometry. All methods use the same initial point which is obtained using the GN-based trilateration method and projecting it onto the equilateral triangle manifold.

All Riemannian algorithms are implemented using the MATLAB toolbox Manopt [37] on an Intel Xeon Processor E5-1650 v4 (15M cache, 3.60 GHz) computer with 32Gb 2.4 GHz DDR4 RAM. In these simulations, the maximum number of iterations is set to 50, the optimality tolerance is set to 10^{-4}, and the step tolerance is set to 10^{-6}.

The size of the room, where the target is located, is given by 30 m x 30 m x 8 m. The locations of the three transmitters are chosen randomly under the constraint that they form an equilateral triangle. The length of the sides of the equilateral triangle, d, is chosen to be 1 cm, unless otherwise indicated. The true ranges from transmitters to beacons, denoted by the d_{ij}'s, are computed and corrupted with a uniformly distributed error e_{ij} to produce the noisy range estimates $r_{ij} = d_{ij} + e_{ij}$. The error, e_{ij}, is bounded by $e_{\max}$, i.e., $-e_{\max} < e_{ij} < e_{\max}$. The value of the maximum error in the range varies from 150 mm to 250 mm with increments of 12.5 mm so as to study multiple scenarios. Typical ultrasound ranging algorithms can achieve such accuracy even under low SNR scenarios [154].

The noisy range estimates r_{ij} are utilized in the GN-based trilateration algorithm to obtain initial estimates of the transmitters locations. These initial locations are projected onto the equilateral triangles manifold by solving (6.13) to obtain an initial point that belongs to the manifold. For each simulation setup, the tests are repeated 1000 times with random transmitters locations and random ranging errors.

The performance of the proposed algorithm is evaluated by comparing the mean square error (MSE) calculated as the average mean square error between the genuine transmitters' position and their estimates. Moreover, the value of the cost function is plotted for each of the algorithms. Furthermore, the computational complexity for the proposed algorithm is compared against the considered benchmark algorithms by calculating the overall running time required to obtain an estimate of the target

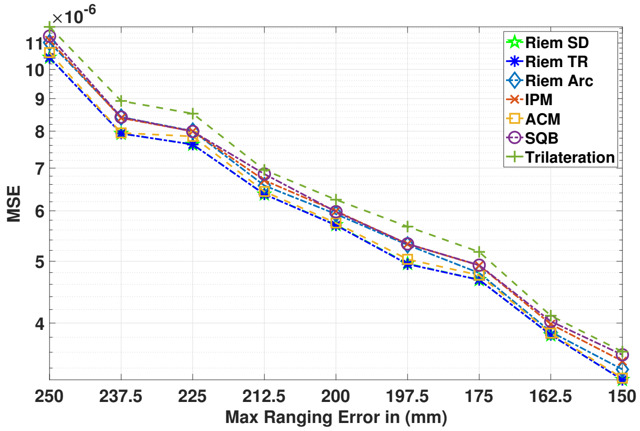


Figure 6.2: Mean square error of the location estimate.

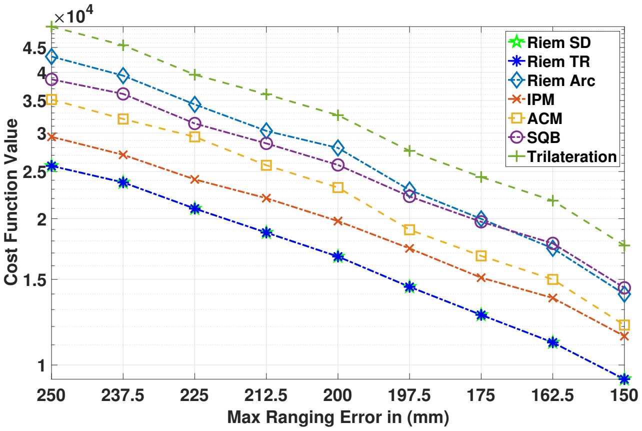


Figure 6.3: Value of the cost function at the reached solution.

position.

Figure 6.2 plots the MSE for each of the tested algorithms. This plot shows the improvement in the localization accuracy when exploiting the geometry of the equilateral triangle of the transmitters as all algorithms have a lower MSE compared to the traditional GN-based trilateration. Moreover, we can see that both the first-order and second-order Riemannian-based localization algorithms outperforms the benchmark algorithms.

Figure 6.3 shows the value of the cost function at the solution obtained by each of the algorithms. The proposed algorithm widely outperforms the benchmark algorithms.

Figure 6.4 shows the cumulative error in the estimated location of the target with a maximum ranging error of 250 mm. More than 90 % of the location estimates are in less than 10^{-5} mm^2 MSE for the proposed Riemannian algorithm. In contrast,

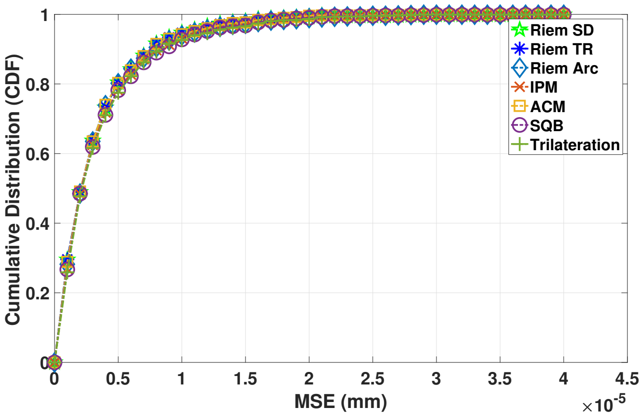


Figure 6.4: CDF of the location error under maximum ranging error of 35 mm.

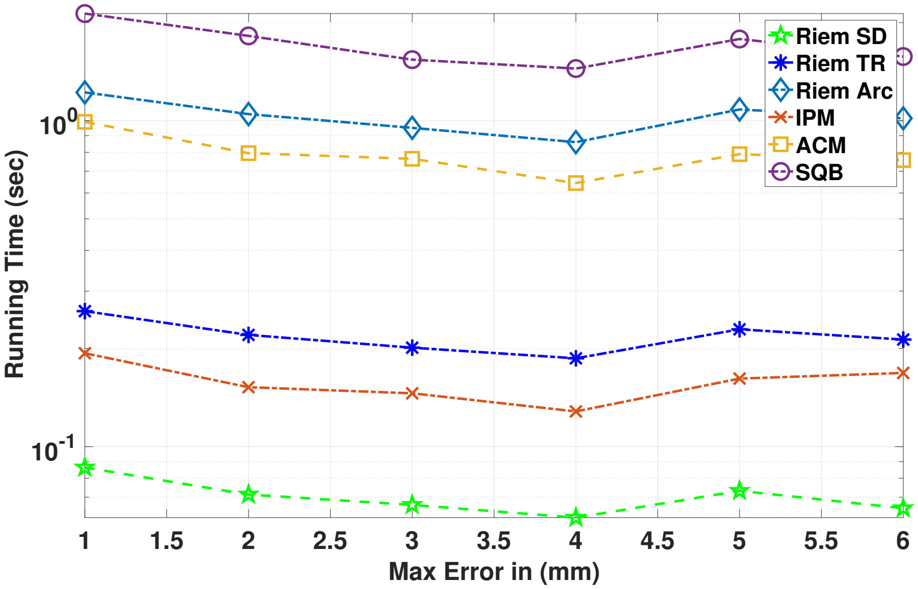


Figure 6.5: Running time to estimate the location of the target.

less than 90 % of the location estimates are in less than 10^{-5} mm^2, 10^{-5} mm^2, and 10^{-5} mm^2 MSE for the IPM, Active Set method and SQP method, respectively.

We compare the computational complexity of all the algorithms by calculating the running time required to reach the minimum of the cost function under different ranging error values. Figure 6.5 shows that the steepest-descent Riemannian localization algorithm requires much lower running time to reach the minimum compared to all benchmark algorithms.

6.2 Precise GNSS Outdoor Attitude Determination

Introduction to Outdoor Localization

The attitude of a vehicle refers to the orientation of its body frame relative to a reference coordinate system [48]. Global Navigation Satellite Systems can provide highly precise 3-D attitude information, which can be determined by two or more non-parallel pointing vectors. GNSS-based attitude determination plays a vital role in vehicle attitude measurement thanks to its high-accuracy, low-cost, stable and real-time performance. As such, it has been widely used in aircrafts, spacecrafts, vessels, automobiles, and many other dynamic platforms [49, 50, 51]. Furthermore, the recent advances in autonomous driving have ignited a renewed interest in GNSS attitude determination, especially for land vehicles.

Although both GNSS pseudo-range and carrier phase measurements can be utilized for attitude estimation, the high precision of GNSS attitude determination mainly derives from the accurate carrier phase measurements. Indeed, the noise level of the carrier phase measurements is a couple of orders of magnitude lower than that of pseudo-range measurements[155]. On the downside, the primary challenge in GNSS attitude determination stems from the fact that the phase observables are ambiguous by an unknown integer number of cycles ambiguity that needs to be resolved. Hence, carrier phase ambiguity resolution, i.e., the process of resolving these unknown integer values, is a crucial step for GNSS attitude determination and other GNSS applications such as positioning, navigation, and so on.

A number of carrier phase ambiguity resolution methods have been proposed over the years, most of which belong to one of the following two classes: motion-based methods and search-based methods [156]. Motion-based methods incorporate information about the dynamics of the vehicle into the estimation process and resolve the carrier phase ambiguities by exploiting the change in the receiver-satellite geometry with time [157, 158, 159]. These methods take advantage of multiple epochs of observations collected over a given period of time assuming that the integer ambiguities remain unchanged during this period. As a consequence of their design, these methods are not well suited for real-time applications [156].

On the other hand, search-based methods, as their name suggests, search for the optimal solution in different domains. Indeed, the search-based methods are more diverse than the motion-based ones thanks to the use of various cost functions and different information provided by the system. For example, there are search-based methods based on carrier phase only or carrier phase combined with pseudo-range, single-frequency versus multi-frequency methods, with or without aid knowledge. As a result of their diversity, these methods differ widely in their efficiency depending on the adopted objective function and the search strategies [160, 161, 162, 163]. For instance, in [161], a constraint equation is used, and the search is carried out in the antenna positions domain. In contrast, the authors in [162] solve the ambiguity using an artificial neural network. For a limited number of GNSS observables, a search algorithm based on Gram-Schmidt orthonormalization (GSO) is developed in [163]. With only four GNSS observables, the algorithm constraints the search space to two dimensions so as to significantly improve the estimation efficiency.

While motion-based methods are not designed for real-time applications, their search-based counterparts are independent of the platform's motion, making them good candidates for instantaneous attitude determination. Among the search-based methods, the LAMBDA approach [160] and its extended versions [164, 165, 166] received considerable attention in the literature. The main advantage of these methods is that the ambiguity resolution search is performed directly in the integer domain. The LAMBDA method solves an integer least-squares problem through an ambiguity transformation that allows for an efficient search for the optimal estimates over a hyper-ellipsoidal region [160].

Depending on the assumptions on the system, GNSS observations can be preprocessed in numerous ways. In particular, the phase observables are processed as phase single, double, or even triple-differences [167]. The phase single-difference, or simply phase difference, is defined as the difference between the simultaneous phase observations (from the same satellite) at two different antennas. The purpose of phase single-difference is to eliminate the satellite clock bias. However, such approach would not solve the problem of receiver clock bias which still needs to be estimated. With common receiver clock technology, single-difference models can be used to simplify the complexity and reduce the influence of uncorrelated noise [168, 169, 170]. The phase double-difference refers to the instantaneous difference between the phase (single) difference observations of two different satellites [171]. By applying double-difference, the clock errors, instrumental delays, atmospheric parameters, and other unknown error sources are significantly lowered to levels that they become almost negligible [172]. As a result, most of the existing GNSS attitude determination methods use the phase double-difference model to carry out baseline settlement. Finally, the phase triple-difference is defined as the difference between the phase double differences from two successive epochs [167]. Assuming that the integer ambiguity remains unchanged during the time interval between epochs, the use of triple difference allows the total removal of the carrier phase integer ambiguity.

In this section, we draw on the ambiguity resolution method proposed in [173, 174, 168] where a particular receiver configuration leads to a simple solution for the ambiguity problem. The search space is reduced to only three candidate integer values, guaranteeing that the search is highly efficient and performs well even with a minimal number of satellites. A single-difference model is adopted with all the antennas operating using a single common clock, which means that the receiver clock bias is automatically canceled [175]. In [173, 174], a triple-antenna configuration is considered, and a solution for the single baseline line pointing direction is developed. This section considers a more complex configuration of five antennas to obtain the pointing direction vectors of non-parallel baselines for more-accurate 3-D attitude determination.

The main contribution of this section is to provide a highly accurate 3-D GNSS attitude determination by rigorously solving the underlying optimization problem. As pointed out in [176], the inclusion of the information about the receiver geometry and baseline length as non-linear constraints in the optimization problem improves the quality of the attitude estimate. While previous works, e.g., [51, 156], use regularization to deal with the non-convexity indirectly, this section proposes a more direct method by solving the non-convex optimization problem using a Riemannian optimization approach. The efficiency of the proposed scheme is tested through extensive simulation with a particular focus on the most challenging case of GNSS attitude determination, i.e., the single-epoch single-frequency case. In contrast to previous works that use Riemannian optimization over well-investigated manifolds to solve the attitude determination problem, e.g., [177, 178], this paper introduces a new manifold and thoroughly investigates its geometry.

Adopting an ambiguity resolution method based on the approach of [173, 174, 168] aims at simplifying the presentation of this paper. Namely, the method of [173, 174, 168] is capable of delivering unambiguous phase-difference estimates after simple algebraic steps. This allows us to focus on the main objective of the paper, and formulate the optimization problem directly using unambiguous phase differences. However, generally speaking, the proposed Riemannian optimization method can be paired with any ambiguity resolution method. This will be demonstrated in the simulation section by pairing the proposed Riemannian optimization method with the MC-LAMBDA method [179].

System Model and Problem Formulation

As stated earlier, this section focuses on the problem of determining the attitude of a platform, equipped with GNSS single-frequency receivers, using the carrier phase measurements. The problem is formulated using the phase differences observed between antenna pairs. Without loss of generality, we assume that the carrier phase and all distances are measured in units of wavelength.

For antenna i, the carrier phase observation equation based on the signal from satellite s is given by

$$\varphi_{i,s} = \rho_{i,s} + \frac{c}{\lambda}(\tau - \tau_s) + I_{i,s} - T_{i,s} + \phi_0 - \phi_{0,s} - n_{i,s} + e_{i,s}, \tag{6.14}$$

where the ambiguous carrier phase $\boldsymbol{\varphi}_{i,s} \in [-0.5, 0.5]$; $\rho_{i,s}$ is the satellite-to-antenna range; c is the speed of light; λ is the wavelength; τ and τ_s are the receiver and satellite clock bias, respectively; $I_{i,s}$ is the ionospheric delay; $T_{i,s}$ is the tropospheric delay; ϕ_0 and $\phi_{0,s}$ are the initial phase offsets for the receiver and satellite at their zero clock time, respectively; $n_{i,s}$ is the unknown integer ambiguity; $e_{i,s}$ includes noise and other errors.

Common receiver clock technology can be realized using the multi-antenna synchronized GNSS receiver. This means that these antennas share identical receiver clock bias and initial phase offset, which can be eliminated by the single-difference operation. The observed phase difference between antenna i and j is given by

$$\varphi_{ij,s} = \varphi_{i,s} - \varphi_{j,s} = \rho_{ij,s} - n_{ij,s} + e_{ij,s}, \tag{6.15}$$

where the atmospheric delays are ignored since the baseline for GNSS attitude determination is usually very short compared to the satellite-to-antenna distance. The unambiguous phase difference $\rho_{ij,s}$ satisfies the following linear relationship:

$$\rho_{ij,s} = d_{ij}\mathbf{h}_s^T\mathbf{x}, \tag{6.16}$$

where d_{ij} denotes to the baseline length, $\mathbf{h}_s$ is the satellite line-of-sight vector, and $\mathbf{x}$ represents the unit direction vector of the antenna baseline.

We consider phase observations from m satellites, and we collect all the phase measurements in a single vector. Under the above conventions and similar to [168], the carrier phase-difference observation equation for attitude determination can be expressed as

$$\boldsymbol{\varphi}_{ij} = d_{ij}\mathbf{H}\mathbf{x} - \mathbf{N}_{ij} + \mathbf{E}, \tag{6.17}$$

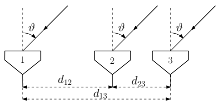


Figure 6.6: Receiver antenna configuration for ambiguity resolution.

where $\mathbf{H}$ is an $m \times 3$ matrix whose rows are the satellite line-of-sight vectors, $\mathbf{N}_{ij} \in \mathbb{Z}^m$ is an integer ambiguity vector, and $\mathbf{E}$ is an unmodeled error vector. The unambiguous carrier phase differences can be modeled as

$$\boldsymbol{\phi}_{ij} = \boldsymbol{\varphi}_{ij} + \mathbf{N}_{ij}. \tag{6.18}$$

Attitude determination can be achieved by estimating two unit vectors (or pointing directions), such as $\mathbf{x}$ in (6.17), using two non-collinear baselines. The integer ambiguity problem, i.e., the fact that the integer vector $\mathbf{N}_{ij}$ in (6.17) is unknown, is the main source of difficulty in applying GNSS carrier phase difference based attitude determination. The resolution of carrier phase ambiguity is a rich research field. This section tailors the phase difference ambiguity resolution method for the specific antenna configuration in [168, 173, 174] to the antenna configuration of interest herein, depicted in Figure 6.6. More details and a rigorous analysis of the integer ambiguity resolution method presented in this section can be found in [173] and [174].

Three collinearly positioned antennas are configured such that the lengths of the shortest two baselines are different, i.e., $d_{12} \neq d_{23}$. The difference between these two baselines, $\triangle = d_{23} - d_{12}$, satisfies the criterion suggested in[174] for ambiguity resolution. In other words, we have the following:

$$0 < |\triangle| \leq \frac{1}{2|\sin(\vartheta)|}, \tag{6.19}$$

wherein $\vartheta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ is the angle of arrival (AoA) of the signal from one satellite through the line-of-sight path, i.e., the angle between the line-of-sight and the plane perpendicular to the antenna baselines. For the sake of simplicity, a fixed value is used in practice. It can readily be noted that when the AoA approaches $\pm\frac{\pi}{2}$, the above inequality simplifies to the following

$$0 < |\triangle| \leq \frac{1}{2}. \tag{6.20}$$

Given the above antenna configuration, there are only three candidate values for the unwrapped phase difference [173]. These candidate phase differences are related to the difference of two carrier phase observables as follows:

$$\boldsymbol{\phi}_{12,s}(k) = \frac{d_{12}}{\triangle}(\boldsymbol{\varphi}_{23,s} - \boldsymbol{\varphi}_{12,s} + k), k = -1, 0, 1, \tag{6.21}$$

where $\boldsymbol{\phi}_{12,s}$ is the unwrapped phase difference between antenna 1 and 2 for the s-th satellite. It is demonstrated in [173] that the desired unwrapped phase difference is the unique one that satisfies $\boldsymbol{\phi}_{12,s}(k) \in [-d_{12}, d_{12}]$. Furthermore, according to the results of [168] and given the unique candidate phase difference, the unique solution of the phase unwrapping problem is the candidate with minimum absolute value, i.e., $\acute{\boldsymbol{\phi}}_{12,s} = \boldsymbol{\phi}_{12,s}(\hat{k})$ for $\hat{k}$ satisfying

$$\hat{k} = \arg \min_{k \in \{-1,0,1\}} |\boldsymbol{\phi}_{12,s}(k)|. \tag{6.22}$$

Finally, due to the impact of noise, the recovered integer ambiguity and the unwrapped phase difference can be refined in practice using

$$\begin{aligned} \hat{\mathbf{n}}_{12,s} &= \lfloor \acute{\boldsymbol{\phi}}_{12,s} - \boldsymbol{\varphi}_{12,s} \rceil, \\ \hat{\boldsymbol{\phi}}_{12,s} &= \boldsymbol{\varphi}_{12,s} + \hat{\mathbf{n}}_{12,s}. \end{aligned} \tag{6.23}$$

This rest of this subsection formulates the 3-D GNSS attitude determination problem. The first part describes the single baseline pointing vector estimation approach. This is done by adapting the least-squares approach proposed in [168] for the configuration of interest herein. Afterward, a second non-collinear baseline pointing vector is considered to obtain precise 3-D GNSS attitude estimation.

This part concerns in determining the single baseline pointing vector which is a unit vector that indicates the pointing direction of the baseline in a reference coordinate system. Given the ambiguity resolution process and the system equations described previously, the single baseline pointing vector determination can be expressed as the following minimization problem

$$\min_{\mathbf{x} \in \mathbb{R}^3} ||\hat{\boldsymbol{\phi}}_{12} - d_{12}\mathbf{H}\mathbf{x}||_2^2, \tag{6.24}$$

which has the least-squares (LS) solution

$$\acute{\mathbf{x}} = \frac{1}{d_{12}}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\hat{\boldsymbol{\phi}}_{12}. \tag{6.25}$$

It can be noted from (6.25) that under the same phase noise, the longer the baseline is, the more-precise the pointing vector is. In other words, the accuracy of the pointing

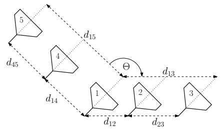


Figure 6.7: Receiver antenna configuration for 3-D attitude determination.

vector estimation is proportional to the length of the antenna baseline. Hence, the longest baseline, d_{13} herein, should be exploited to acquire a more-precise pointing vector which can be realized using the upcoming operations

$$\begin{aligned}\hat{\mathbf{n}}_{13} &= \lfloor d_{13}\mathbf{H}\acute{\mathbf{x}} - \boldsymbol{\varphi}_{13}\rceil,\\ \hat{\boldsymbol{\phi}}_{13} &= \boldsymbol{\varphi}_{13} + \hat{\mathbf{n}}_{13},\\ \hat{\mathbf{x}} &= \frac{1}{d_{13}}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T\hat{\boldsymbol{\phi}}_{13}.\end{aligned} \tag{6.26}$$

wherein $\lfloor . \rceil$ refers to the rounding operation.

Single pointing vector estimation is the first step in 3-D attitude measurement. Indeed, in order to obtain 3-D attitude information, one needs at least another baseline which is not parallel to the first one. A natural extension of the configuration in Figure 6.6 that can tackle the 3-D attitude determination and ambiguity resolution is shown in Figure 6.7. One antenna is shared by the two baselines so as to reduce the number of required antennas. As a consequence, only 5 antennas are required instead of 6. The angle between the two baseline directions is denoted by Θ. Similar to the first baseline direction, the difference between the two shorter baselines of the antenna configuration of antenna 1, 4, and 5 satisfies the criterion

$$0 < |\triangle'| = |d_{45} - d_{14}| \leq \frac{1}{2}, \tag{6.27}$$

In a similar fashion as for (6.24)-(6.26), the carrier phase difference vectors $\hat{\boldsymbol{\phi}}_{14}$ and $\hat{\boldsymbol{\phi}}_{15}$, the integer ambiguity vector $\hat{\mathbf{n}}_{14}$ and $\hat{\mathbf{n}}_{15}$, and the corresponding pointing vector $\hat{\mathbf{y}}$ can be estimated. In other words, the two baseline pointing vectors are estimated independently by solving the minimization problem (6.24) for each of the two baselines separately. Once the two baseline vectors have been estimated, two non-parallel pointing vectors are known in the reference coordinate and the 3-D attitude of platform can be calculated by transferring these two pointing vectors to an attitude matrix in a straightforward manner. However, the described method does not take the antenna geometry and baseline length into consideration which can greatly reduce the precision of the estimation. We suggest instead to use two baselines jointly to incorporate all the available information. After recovering the unambiguous phase differences, the 3-D GNSS attitude determination problem can be formulated as

$$\min_{\mathbf{x},\mathbf{y}\in\mathbb{R}^3} ||\boldsymbol{\phi}_{12} - d_{12}\mathbf{H}\mathbf{x}||_2^2 + ||\boldsymbol{\phi}_{14} - d_{14}\mathbf{H}\mathbf{y}||_2^2 \tag{6.28a}$$

$$\text{s.}t.\ ||\mathbf{x}||_2^2 = 1 \tag{6.28b}$$

$$||\mathbf{y}||_2^2 = 1 \tag{6.28c}$$

$$\langle \mathbf{x}, \mathbf{y} \rangle = \cos(\Theta). \tag{6.28d}$$

This is a non-convex optimization problem as indicated by the three quadratic equality constraints. More generally, let **a** and **b** be two m-dimensional vectors, let **A** and **B** be two $m \times n$ matrices and consider a scalar $-1 < c < 1$. This paper focuses on solving the following optimization problem:

$$\min_{\mathbf{x},\mathbf{y}\in\mathbb{R}^n} ||\mathbf{a} - \mathbf{A}\mathbf{x}||_2^2 + ||\mathbf{b} - \mathbf{B}\mathbf{y}||_2^2 \tag{6.29a}$$

$$\text{s.}t.\ ||\mathbf{x}||_2^2 = 1 \tag{6.29b}$$

$$||\mathbf{y}||_2^2 = 1 \tag{6.29c}$$

$$\langle \mathbf{x}, \mathbf{y} \rangle = c, \tag{6.29d}$$

wherein constraint (6.29b) underlines the fact that **x** is a unit norm vector and constraint (6.29c) corresponds to the fact that **y** is also a unit norm vector. Finally, constraint (6.29d) insists that the cosine of the angle between **x** and **y** is c. Notice that the strict inequalities in $-1 < c < 1$ comes without loss of generality. Indeed, if we set $c = 1$, then $\mathbf{x} = \mathbf{y}$ which allows to reformulate the optimization problem in a convex, single-variable program. The same is valid for $c = -1$ for which $\mathbf{x} = -\mathbf{y}$.

Despite the convexity of the objective function (6.29a), the optimization problem in (6.29) is not convex due to the coupling of variables in constraint (6.29d). However, the set $\mathcal{M}$ of potential solutions given by

$$\mathcal{M} = \left\{(\mathbf{x}, \mathbf{y}) \in \left(\mathbb{R}^n\right)^2 \mid \mathbf{x}^{\mathrm{T}}\mathbf{x} = \mathbf{y}^{\mathrm{T}}\mathbf{y} = 1, \mathbf{x}^{\mathrm{T}}\mathbf{y} = c\right\} \tag{6.30}$$

forms a compact, i.e., closed and bounded, manifold embedded in the Euclidean space $(\mathbb{R}^n)^2 = \mathbb{R}^n \times \mathbb{R}^n$. Therefore, one can takes advantage of Riemannian optimization algorithms over manifolds to efficiently solve the above problem. Indeed, thanks to the compactness of the manifold $\mathcal{M}$ and the convexity of the objective function (6.29a), first-order Riemannian methods are guaranteed to converge to a critical point of the optimization problem [23], i.e., a saddle point, local minimum or local maximum. However, it has been observed in the literature that first-order methods converge to a local minimum unless the initialization is specifically crafted (see [23]). On the other hand, second-order methods are always guaranteed to converge to a local minimum due to the instability of saddle points and local maxima for these methods [23]. This thesis solves the non-convex optimization problem (6.29) by designing both first and second-order Riemannian algorithms over the manifold $\mathcal{M}$, called herein the oriented spheres manifold.

Optimization Over the Set of Unit Norm Oriented Vectors

As stated previously, the tangent space can be computed using the implicit function theorem available in Theorem 5.2. The following theorem derives the expression of the tangent bundle $\mathcal{TM}$ of the oriented spheres manifold $\mathcal{M}$.

Theorem 6.1 *The tangent space $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ at the point $(\mathbf{x},\mathbf{y}) \in \mathcal{M}$ is given by the following $(2n-3)$-dimensional Euclidean space*

$$\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M} = \Big\{(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) \in \left(\mathbb{R}^n\right)^2 \mid \mathbf{x}^{\mathrm{T}}\xi_{\mathbf{x}} = \mathbf{y}^{\mathrm{T}}\eta_{\mathbf{y}} = 0, \\ \mathbf{x}^{\mathrm{T}}\eta_{\mathbf{y}} + \mathbf{y}^{\mathrm{T}}\xi_{\mathbf{x}} = 0\Big\}. \qquad (6.31)$$

Proof: Let $\mathcal{S}^n$ be the set of $n \times n$ symmetric matrices and consider the function $\mathsf{F} : (\mathbb{R}^n)^2 \to \mathcal{S}^2$ given by

$$\mathsf{F}(\mathbf{x},\mathbf{y}) = \begin{pmatrix} \mathbf{x}^{\mathrm{T}}\mathbf{x} & \mathbf{x}^{\mathrm{T}}\mathbf{y} \\ \mathbf{y}^{\mathrm{T}}\mathbf{x} & \mathbf{y}^{\mathrm{T}}\mathbf{y}. \end{pmatrix} \qquad (6.32)$$

The set of symmetric matrices $\mathcal{S}^n$ is a linear set as mandated by the conditions of Theorem 5.2. Furthermore, from the definition of the function F, it is clear that the oriented spheres manifold $\mathcal{M}$ is the level set of $\mathbf{C} = \begin{pmatrix} 1 & c \\ c & 1 \end{pmatrix}$ by the continuous and smooth function F. Indeed, the manifold can be expressed as $\mathcal{M} = \mathsf{F}^{-1}(\mathbf{C})$. In order to show that the map F is a constant-rank function, it is sufficient to show that $\mathbf{C}$ is a regular value of F, i.e., the rank of each $(\mathbf{x},\mathbf{y}) \in \mathsf{F}^{-1}(\mathbf{C}) = \mathcal{M}$ is equal to $\mathrm{Dim}(\mathcal{S}^2) = \frac{2(2+1)}{2} = 3$. In other words, we need to show that the indefinite directional derivative of F is a surjective map.

The directional derivative of F at $(\mathbf{x},\mathbf{y})$ in the direction $(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) \in (\mathbb{R}^n)^2$ is given by

$$\mathsf{D}\left(\mathsf{F}(\mathbf{x},\mathbf{y})\right)[\xi_{\mathbf{x}}, \eta_{\mathbf{y}}] = \begin{pmatrix} 2\mathbf{x}^{\mathrm{T}}\xi_{\mathbf{x}} & \mathbf{x}^{\mathrm{T}}\eta_{\mathbf{y}} + \mathbf{y}^{\mathrm{T}}\xi_{\mathbf{x}} \\ \mathbf{x}^{\mathrm{T}}\eta_{\mathbf{y}} + \mathbf{y}^{\mathrm{T}}\xi_{\mathbf{x}} & 2\mathbf{y}^{\mathrm{T}}\eta_{\mathbf{y}}. \end{pmatrix} \qquad (6.33)$$

Let $\mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{pmatrix} \in \mathcal{S}^n$ be an arbitrary 2×2 symmetric matrix, i.e., $\mathbf{S}_{21} = \mathbf{S}_{12}$. Finding $(\xi_{\mathbf{x}}, \eta_{\mathbf{y}})$ such that $\mathsf{D}\left(\mathsf{F}(\mathbf{x}, \mathbf{y})\right)[\xi_{\mathbf{x}}, \eta_{\mathbf{y}}] = \mathbf{S}$ can be reduced to solving the following linear system of equations

$$\mathbf{A} \begin{pmatrix} \xi_{\mathbf{x}} \\ \eta_{\mathbf{y}} \end{pmatrix} = \begin{pmatrix} 2\mathbf{x}^{\mathrm{T}} & \mathbf{0} \\ \mathbf{0} & 2\mathbf{y}^{\mathrm{T}} \\ \mathbf{y}^{\mathrm{T}} & \mathbf{x}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \xi_{\mathbf{x}} \\ \eta_{\mathbf{y}} \end{pmatrix} = \begin{pmatrix} \mathbf{S}_{11} \\ \mathbf{S}_{22} \\ \mathbf{S}_{12} \end{pmatrix}. \tag{6.34}$$

With the obvious assumption that $n \geq 2$, the linear system has a fat matrix $\mathbf{A}$ with dimension $3 \times 2n$. Furthermore, the matrix $\mathbf{A}$ is full-row rank. Indeed, assume that some linear combination of the rows with the scalars α, β, and γ, respectively, gives a zero vector. In other words, we have the following system of equations:

$$2\alpha\mathbf{x}^{\mathrm{T}} + \gamma\mathbf{y}^{\mathrm{T}} = \mathbf{0}^{\mathrm{T}}, \tag{6.35}$$

$$2\beta\mathbf{y}^{\mathrm{T}} + \gamma\mathbf{x}^{\mathrm{T}} = \mathbf{0}^{\mathrm{T}}. \tag{6.36}$$

Multiplying each equation in the above system of equations by $\mathbf{x}$ and $\mathbf{y}$, and rearranging the expressions yields

$$\begin{pmatrix} 2 & 0 & c \\ 2c & 0 & 1 \\ 0 & 2 & c \\ 0 & 2c & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{6.37}$$

To attest that the matrix above has trivial Null space, one can notice that the determinant of the first 3×3 block is $4c^2 - 4$ which does not have solutions for $c \in (-1, 1)$. Therefore, we conclude that the matrix $\mathbf{A}$ is full-row rank and thus the map $\mathsf{D}\left(\mathsf{F}(\mathbf{x}, \mathbf{y})\right)$ is surjective which concludes that F is a rank-constant function.

Exploiting the result of Theorem 5.2, the tangent space of $\mathcal{M}$ can be expressed as the set of directions that annihilate the directional derivative of the function F. In other words, we obtain the following characterization of the tangent space

$$\begin{aligned} \mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M} &= \operatorname{Ker}(\mathsf{D}\left(\mathsf{F}(\mathbf{x}, \mathbf{y})\right)) \\ &= \left\{(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) \in \left(\mathbb{R}^n\right)^2 \mid \mathsf{D}\left(F(\mathbf{x}, \mathbf{y})\right)[\xi_{\mathbf{x}}, \eta_{\mathbf{y}}] = \mathbf{0}\right\} \\ &= \Big\{(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) \in \left(\mathbb{R}^n\right)^2 \mid \mathbf{x}^{\mathrm{T}}\xi_{\mathbf{x}} = \mathbf{y}^{\mathrm{T}}\eta_{\mathbf{y}} = 0, \\ &\qquad\qquad \mathbf{x}^{\mathrm{T}}\eta_{\mathbf{y}} + \mathbf{y}^{\mathrm{T}}\xi_{\mathbf{x}} = 0\Big\}. \end{aligned} \tag{6.38}$$

Finally, assuming $n \geq 2$ and using the submersion theorem of Riemannian manifold [23], we conclude that $\mathcal{M} = \mathsf{F}^{-1}(\mathbf{C})$ is a manifold of dimension $\text{Dim}((\mathbb{R}^n)^2) - \text{Dim}(\mathcal{S}^2) = 2n - 3$. The dimension of the manifold can also be concluded from the expression of the tangent space derived above. ■

Let the embedding space $(\mathbb{R}^n)^2$ be equipped with the Frobenius inner product, defined as $\langle(\mathbf{x}, \mathbf{y}), (\mathbf{x}', \mathbf{y}')\rangle = \mathbf{x}^{\mathrm{T}}\mathbf{x}' + \mathbf{y}^{\mathrm{T}}\mathbf{y}'$ for all vectors $(\mathbf{x}, \mathbf{y})$ and $(\mathbf{x}', \mathbf{y}')$ in $((\mathbb{R}^n)^2)$. This section considers that the embedded manifold inherits the inner product of the embedding space. In other words, the induced inner product $\langle ., . \rangle_{(\mathbf{x},\mathbf{y})}$ on the tangent space $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ for $(\mathbf{x}, \mathbf{y}) \in \mathcal{M}$ is given by

$$\langle(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}), (\xi'_{\mathbf{x}}, \eta'_{\mathbf{y}})\rangle_{(\mathbf{x},\mathbf{y})} = \xi_{\mathbf{x}}^T \xi'_{\mathbf{x}} + \eta_{\mathbf{y}}^T \eta'_{\mathbf{y}}, \tag{6.39}$$

for all tangent vectors $(\xi_{\mathbf{x}}, \eta_{\mathbf{y}})$ and $(\xi'_{\mathbf{x}}, \eta'_{\mathbf{y}})$ in $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$.

As this section considers the induced inner product from the ambient space $(\mathbb{R}^n)^2$, the expression of the Riemannian metric simplifies to $\mathsf{grad}\,\mathsf{f}(\mathbf{x}, \mathbf{y}) = \Pi_{(\mathbf{x},\mathbf{y})}(\mathsf{Grad}\,\mathsf{f}(\mathbf{x}, \mathbf{y}))$ with $\Pi_{(\mathbf{x},\mathbf{y})} : (\mathbb{R}^n)^2 \rightarrow \mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ being the orthogonal projection from the ambient space to the tangent one [74, 75]. In the rest of this section, the first and second components of the Riemannian gradient are denoted by $\mathsf{grad}_{\mathbf{x}}\mathsf{f}(\mathbf{x}, \mathbf{y})$ and $\mathsf{grad}_{\mathbf{y}}\mathsf{f}(\mathbf{x}, \mathbf{y})$, respectively. The same notation is used for the Euclidean gradient, i.e., $\mathsf{Grad}_{\mathbf{x}}\mathsf{f}(\mathbf{x}, \mathbf{y})$ and $\mathsf{Grad}_{\mathbf{y}}\mathsf{f}(\mathbf{x}, \mathbf{y})$ for the derivative with respect to $\mathbf{x}$ and $\mathbf{y}$, respectively. The following theorem relates the expression of the Riemannian gradient $\mathsf{grad}\,\mathsf{f}(\mathbf{x}, \mathbf{y}) = \Pi_{(\mathbf{x},\mathbf{y})}(\mathsf{Grad}\,\mathsf{f}(\mathbf{x}, \mathbf{y}))$ to its Euclidean counterpart $\mathsf{Grad}\,\mathsf{f}(\mathbf{x}, \mathbf{y})$ by providing the expression of the orthogonal projection $\Pi_{(\mathbf{x},\mathbf{y})}$.

Theorem 6.2 *The orthogonal projection $\Pi_{(\mathbf{x},\mathbf{y})}$ from the ambient space $\left(\mathbb{R}_*^n\right)^2$ to the tangent space $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ is given by*

$$\Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}) = \begin{pmatrix} \mathbf{u} - \dfrac{1}{2(1-c^2)}(\alpha\mathbf{x} + \gamma\mathbf{y}) \\ \mathbf{v} - \dfrac{1}{2(1-c^2)}(\gamma\mathbf{x} + \beta\mathbf{y}). \end{pmatrix} \tag{6.40}$$

where the scalars α, β, and γ are given by the following equations

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} (2-c^2)\mathbf{x}^{\mathrm{T}}\mathbf{u} + c^2\mathbf{y}^{\mathrm{T}}\mathbf{v} - c(\mathbf{x}^{\mathrm{T}}\mathbf{v} + \mathbf{y}^{\mathrm{T}}\mathbf{u}) \\ (2-c^2)\mathbf{y}^{\mathrm{T}}\mathbf{v} + c^2\mathbf{x}^{\mathrm{T}}\mathbf{u} - c(\mathbf{y}^{\mathrm{T}}\mathbf{u} + \mathbf{x}^{\mathrm{T}}\mathbf{v}) \\ \mathbf{x}^{\mathrm{T}}(\mathbf{v} - c\mathbf{u}) + \mathbf{y}^{\mathrm{T}}(\mathbf{u} - c\mathbf{v}) \end{pmatrix}. \tag{6.41}$$

Proof: In order to derive the expression of the Riemannian gradient, one needs to compute the orthogonal projection from the ambient space to the tangent bundle. Such orthogonal projection is obtained by first expressing the orthogonal complement of the tangent space in the below lemma.

Lemma 6.1 *The orthogonal complement of* $\mathcal{T}^{\perp}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ *of the tangent space* $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ *at* $(\mathbf{x},\mathbf{y}) \in \mathcal{M}$ *is given by*

$$\begin{aligned}\mathcal{T}^{\perp}_{(\mathbf{x},\mathbf{y})}\mathcal{M} = \Big\{(\zeta_{\mathbf{x}},\psi_{\mathbf{y}}) \in \left(\mathbb{R}^n\right)^2 \mid \zeta_{\mathbf{x}} &= \alpha\mathbf{x} + \gamma\mathbf{y} \\ \psi_{\mathbf{y}} &= \gamma\mathbf{x} + \beta\mathbf{y}, \alpha, \beta, \gamma \in \mathbb{R}\Big\}.\end{aligned} \tag{6.42}$$

Proof: The proof of this lemma is obtained by double inclusion of the orthogonal complement of the tangent space and the proposed set in the lemma. The first inclusion is asserted by direct computation of the inner product between the tangent space and its orthogonal complement. A dimension counting argument concludes the proof.

Recall that the tangent vector $(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) \in \mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ satisfies the following equality

$$\begin{aligned}\mathbf{x}^{\mathrm{T}}\xi_{\mathbf{x}} &= \mathbf{y}^{\mathrm{T}}\eta_{\mathbf{y}} = 0, \\ \mathbf{x}^{\mathrm{T}}\eta_{\mathbf{y}} + \mathbf{y}^{\mathrm{T}}\xi_{\mathbf{x}} &= 0.\end{aligned} \tag{6.43}$$

Now, consider the vector $(\zeta_{\mathbf{x}}, \psi_{\mathbf{y}})$ satisfying $\zeta_{\mathbf{x}} = \alpha\mathbf{x} + \gamma\mathbf{y}$ and $\psi_{\mathbf{y}} = \gamma\mathbf{x} + \beta\mathbf{y}$ for some real numbers α, β, and γ. The inner product can be expressed as

$$\begin{aligned}\langle(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}), (\zeta_{\mathbf{x}}, \psi_{\mathbf{y}})\rangle_{(\mathbf{x},\mathbf{y})} &= \xi_{\mathbf{x}}^{T}\zeta_{\mathbf{x}} + \eta_{\mathbf{y}}^{T}\psi_{\mathbf{y}} \\ &= \xi_{\mathbf{x}}^{T}(\alpha\mathbf{x} + \gamma\mathbf{y}) + \eta^{T}_{\mathbf{y}(}\gamma\mathbf{x} + \beta\mathbf{y}) \\ &= \gamma(\xi_{\mathbf{x}}^{\mathrm{T}}\mathbf{y} + \eta_{\mathbf{y}}^{\mathrm{T}}\mathbf{x}) = 0.\end{aligned} \tag{6.44}$$

Therefore, we obtain the inclusion

$$\begin{aligned}\mathcal{T}^{\perp}_{(\mathbf{x},\mathbf{y})}\mathcal{M} \subseteq \Big\{(\zeta_{\mathbf{x}},\psi_{\mathbf{y}}) \in \left(\mathbb{R}^n\right)^2 \mid \zeta_{\mathbf{x}} &= \alpha\mathbf{x} + \gamma\mathbf{y} \\ \psi_{\mathbf{y}} &= \gamma\mathbf{x} + \beta\mathbf{y}, \alpha, \beta, \gamma \in \mathbb{R}\Big\}.\end{aligned} \tag{6.45}$$

Now notice that the dimension of the ambient space is $2n$ and the dimension of the tangent space is $2n - 3$ which concludes that $\mathcal{T}^{\perp}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ is of dimension 3 which matches the dimension of the set in the left-hand side of (6.42). Therefore, we conclude the equality of both sets. ■

With the characterization of the orthogonal complement of the tangent space in the previous lemma, we now compute the orthogonal projection from the ambient space to the tangent one. Consider a vector $(\mathbf{u}, \mathbf{v}) \in (\mathbb{R}^n)^2$ in the ambient space. Such vector can be decomposed into a tangent and an orthogonal component as follows:

$$(\mathbf{u}, \mathbf{v}) = \Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}) + \Pi^{\perp}_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}), \tag{6.46}$$

where $\Pi^{\perp}_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v})$ is the projection onto the orthogonal complement of the tangent space that can be expressed as

$$\Pi^{\perp}_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}) = \begin{pmatrix} \alpha\mathbf{x} + \gamma\mathbf{y} \\ \gamma\mathbf{x} + \beta\mathbf{y} \end{pmatrix} \tag{6.47}$$

for some real numbers α, β, and γ. Recall that the tangent vector $\Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v})$ satisfies the following three equations

$$\begin{aligned} \langle \Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}), (\mathbf{x}, \mathbf{0}) \rangle_{(\mathbf{x},\mathbf{y})} &= \mathbf{0} \\ \langle \Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}), (\mathbf{0}, \mathbf{y}) \rangle_{(\mathbf{x},\mathbf{y})} &= \mathbf{0} \\ \langle \Pi_{(\mathbf{x},\mathbf{y})}(\mathbf{u}, \mathbf{v}), (\mathbf{y}, \mathbf{x}) \rangle_{(\mathbf{x},\mathbf{y})} &= \mathbf{0} \end{aligned} \tag{6.48}$$

Combining the equations in (6.46) with the tangent space characterization in (6.48) and using the expression of the orthogonal projection in (6.40), we obtain the following three equations

$$\mathbf{x}^{\mathrm{T}}(\alpha\mathbf{x} + \gamma\mathbf{y}) = \mathbf{x}^{\mathrm{T}}\mathbf{u} \tag{6.49}$$

$$\mathbf{y}^{\mathrm{T}}(\gamma\mathbf{x} + \beta\mathbf{y}) = \mathbf{y}^{\mathrm{T}}\mathbf{v} \tag{6.50}$$

$$\mathbf{y}^{\mathrm{T}}(\alpha\mathbf{x} + \gamma\mathbf{y}) + \mathbf{x}^{\mathrm{T}}(\gamma\mathbf{x} + \beta\mathbf{y}) = \mathbf{x}^{\mathrm{T}}\mathbf{v} + \mathbf{y}^{\mathrm{T}}\mathbf{u}. \tag{6.51}$$

Now using the manifold equations, the above equations can be simplified to the following linear system in the variables α, β, and γ.

$$\begin{pmatrix} 1 & 0 & c \\ 0 & 1 & c \\ c & c & 2 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} \mathbf{x}^{\mathrm{T}}\mathbf{u} \\ \mathbf{y}^{\mathrm{T}}\mathbf{v} \\ \mathbf{x}^{\mathrm{T}}\mathbf{v} + \mathbf{y}^{\mathrm{T}}\mathbf{u} \end{pmatrix}. \tag{6.52}$$

Finally, noticing that

$$\begin{pmatrix} 1 & 0 & c \\ 0 & 1 & c \\ c & c & 2 \end{pmatrix}^{-1} = \frac{1}{2(1 - c^2)} \begin{pmatrix} 2 - c^2 & c^2 & -c \\ c^2 & 2 - c^2 & -c \\ -c & -c & 1 \end{pmatrix}, \tag{6.53}$$

and rearranging the terms concludes the proof. ■

The fact that the manifold $\mathcal{M}$ is an embedded manifold in a Euclidean ambient space allows us to simplify the expression of the Riemannian Hessian. Indeed, as stated earlier, the expression of the Riemannian Hessian can be simplified to a directional derivative followed by an orthogonal projection, i.e.,

$$\begin{aligned} \mathsf{hess}\ \mathsf{f}(\mathbf{x},\mathbf{y})[\xi_{\mathbf{x}},\eta_{\mathbf{y}}] &= \Pi_{(\mathbf{x},\mathbf{y})}(\mathsf{D}\ (\mathsf{grad}\ \mathsf{f}(\mathbf{x},\mathbf{y}))[\xi_{\mathbf{x}},\eta_{\mathbf{y}}]) \\ &= \Pi_{(\mathbf{x},\mathbf{y})} \begin{pmatrix} \mathsf{D}\ (\mathsf{grad}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x},\mathbf{y}))[\xi_{\mathbf{x}},\eta_{\mathbf{y}}] \\ \mathsf{D}\ (\mathsf{grad}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x},\mathbf{y}))[\xi_{\mathbf{x}},\eta_{\mathbf{y}}] \end{pmatrix}. \end{aligned} \tag{6.54}$$

Using the above characterization of the Riemannian Hessian, the next proposition establishes the relation between the Riemannian Hessian and its Euclidean Hessian counterpart.

Proposition 6.1 *The Riemannian Hessian (denoted by* $\mathsf{hess}\ \mathsf{f}(\mathbf{x},\mathbf{y})[\xi_{\mathbf{x}},\eta_{\mathbf{y}}]$*) of the function* f *at the point* $(\mathbf{x},\mathbf{y}) \in \mathcal{M}$ *and in the direction* $[\xi_{\mathbf{x}},\eta_{\mathbf{y}}]$ *is given by the following expression*

$$\mathsf{hess}\ \mathsf{f}(\mathbf{x},\mathbf{y}) = \begin{pmatrix} \mathsf{hess}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x},\mathbf{y}) \\ \mathsf{hess}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x},\mathbf{y}) \end{pmatrix} = \Pi_{(\mathbf{x},\mathbf{y})} \begin{pmatrix} \overline{\Delta}_{\mathbf{x}} \\ \overline{\Delta}_{\mathbf{y}} \end{pmatrix}, \tag{6.55}$$

where the components of the vector are given by

$$\overline{\Delta}_{\mathbf{x}} = \Delta_{\mathbf{x}} - \frac{1}{2(1-c^2)} \left(\dot{\alpha}\mathbf{x} + \alpha\xi_{\mathbf{x}} + \dot{\gamma}\mathbf{y} + \gamma\eta_{\mathbf{y}} \right), \tag{6.56}$$

$$\overline{\Delta}_{\mathbf{y}} = \Delta_{\mathbf{y}} - \frac{1}{2(1-c^2)} \left(\dot{\gamma}\mathbf{x} + \gamma\xi_{\mathbf{x}} + \dot{\beta}\mathbf{y} + \beta\eta_{\mathbf{y}} \right). \tag{6.57}$$

The scalars α*,* β*, and* γ *are given by*

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \mathbf{A} \begin{pmatrix} \mathbf{x}^{\mathrm{T}}\mathsf{Grad}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x},\mathbf{y}) \\ \mathbf{y}^{\mathrm{T}}\mathsf{Grad}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x},\mathbf{y}) \\ \mathbf{x}^{\mathrm{T}}\mathsf{Grad}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x},\mathbf{y}) + \mathbf{y}^{\mathrm{T}}\mathsf{Grad}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x},\mathbf{y}) \end{pmatrix}, \tag{6.58}$$

where $\mathbf{A}$ *is the constant matrix* $\begin{pmatrix} 2-c^2 & c^2 & -c \\ c^2 & 2-c^2 & -c \\ -c & -c & 1 \end{pmatrix}$ *and the derivatives* $\dot{\alpha}$*,* $\dot{\beta}$*, and*

$\dot{\gamma}$ are given by the following equations:

$$\begin{pmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{pmatrix} = \mathbf{A} \begin{pmatrix} \xi_{\mathbf{x}}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{x}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} \\ \mathbf{0}^{\mathrm{T}} & \eta_{\mathbf{y}}^{\mathrm{T}} & \mathbf{0}^{\mathrm{T}} & \mathbf{y}^{\mathrm{T}} \\ \eta_{\mathbf{y}}^{\mathrm{T}} & \xi_{\mathbf{x}}^{\mathrm{T}} & \mathbf{y}^{\mathrm{T}} & \mathbf{x}^{\mathrm{T}} \end{pmatrix} \begin{pmatrix} \mathsf{Grad}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x}, \mathbf{y}) \\ \mathsf{Grad}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x}, \mathbf{y}) \\ \mathsf{Hess}\ _{\mathbf{x}}\mathsf{f}(\mathbf{x}, \mathbf{y})\,[\xi_{\mathbf{x}}, \eta_{\mathbf{y}}] \\ \mathsf{Hess}\ _{\mathbf{y}}\mathsf{f}(\mathbf{x}, \mathbf{y})\,[\xi_{\mathbf{x}}, \eta_{\mathbf{y}}] \end{pmatrix}. \tag{6.59}$$

Proof: The proof of this proposition is omitted herein as it follows directly from the expression of the Riemannian gradient and the definition of the Riemannian Hessian for a manifold $\mathcal{M}$ embedded in a Euclidean space endowed with the induced canonical inner product, i.e., the Riemannian Hessian characterization in (6.54) and the Riemannian gradient expression in Theorem 6.2. ■

To design a computationally efficient retraction $\mathsf{R}_{(\mathbf{x},\mathbf{y})}$ on the tangent space $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ of the manifold $\mathcal{M}$, this section exploits Theorem 3.2 as the manifold is defined by only equality constraints. Let $\mathcal{N} = \mathbb{R}_* \times \mathbb{R}_* \times \mathbb{R}$ be a manifold of dimension 3. Therefore, we have $\dim(\mathcal{M}) + \dim(\mathcal{N}) = \dim((\mathbb{R}^n)^2)$. Now consider the mapping ϕ defined by

$$\begin{aligned} \phi : \qquad & \mathcal{M} \times \mathcal{N} \longrightarrow \mathcal{E}^* \subset \left(\mathbb{R}^n\right)^2 \\ & \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} \times \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \longmapsto \begin{pmatrix} \alpha\mathbf{x} + \gamma\mathbf{y} \\ \gamma\mathbf{x} + \beta\mathbf{y} \end{pmatrix}, \end{aligned} \tag{6.60}$$

with $\mathcal{E}^* = \left(\mathbb{R}_*^n\right)^2$ representing the open subset of $(\mathbb{R}^n)^2$ containing only non-zero vectors in each of the two components. Note that the function ϕ is a continuous and differentiable function for all input parameters. Furthermore, for any input parameters $(\mathbf{x}, \mathbf{y}) \in \mathcal{M}$, notice that

$$\phi\left(\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}\right) = \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}. \tag{6.61}$$

In other words, the element $\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \in \mathcal{N}$ is a neutral element for ϕ as mandated by Theorem 3.2. Finally, the following theorem provides the expression of the proposed retraction.

Theorem 6.3 *The mapping* $\mathsf{R} : \mathcal{TM} \to \mathcal{M}$ *whose restriction* $\mathsf{R}_{(\mathbf{x},\mathbf{y})}$ *to* $\mathcal{T}_{(\mathbf{x},\mathbf{y})}\mathcal{M}$ *is given by*

$$\mathsf{R}_{(\mathbf{x},\mathbf{y})}(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) = (\mathbf{v}, \mathbf{w}), \tag{6.62}$$

with the vector $(\mathbf{v}, \mathbf{w})$ *being expressed as*

$$\begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix} = \begin{pmatrix} \cos(\theta)\mathbf{I}_n & -\sin(\theta)\mathbf{I}_n \\ \cos(\varphi)\mathbf{I}_n & \sin(\varphi)\mathbf{I}_n \end{pmatrix} \begin{pmatrix} \mathbf{E}_1 \\ \mathbf{e}_2 \end{pmatrix} \tag{6.63}$$

where the angles θ *and* φ*, the basis vectors* $\mathbf{e}_1$ *and* $\mathbf{e}_2$*, and the constants* c_1*,* c_2*, and* c_3 *are given by*

$$\begin{aligned}
\theta &= \cos^{-1}(c) - \varphi \\
\varphi &= \tan^{-1}\left(\frac{c_1\sqrt{1-c^2} + c_3}{c_2 + c_1 c}\right) \\
c_1 &= ||\mathbf{x} + \xi_{\mathbf{x}}||_2 \\
\mathbf{e}_1 &= \frac{\mathbf{x} + \xi_{\mathbf{x}}}{c_1} \\
c_2 &= (\mathbf{y} + \eta_{\mathbf{y}})^{\mathrm{T}}\mathbf{e}_1 \\
c_3 &= ||(\mathbf{y} + \eta_{\mathbf{y}}) - c_2\mathbf{e}_1||_2 \\
\mathbf{e}_2 &= \frac{(\mathbf{y} + \eta_{\mathbf{y}}) - c_2\mathbf{e}_1}{c_3}.
\end{aligned}$$

Proof: This theorem is demonstrated by applying the result of Theorem 3.2 to the mapping ϕ. Given the preliminary results above, the expression of the retraction is proven by showing that ϕ is a diffeomorphism and by computing its inverse. In other words, the retraction is derived by showing that ϕ is a bijection from $\mathcal{M} \times \mathcal{N}$ to $\mathcal{E}^* = \phi(\mathcal{M}, \mathcal{N})$ and by obtaining the expression of the inverse.

Let $(\mathbf{v}, \mathbf{w})$ with $\mathbf{v} \neq \mathbf{0}$ and $\mathbf{w} \neq \mathbf{0}$ be a vector in $\mathcal{E}^*$. It can readily be seen that, up to the permutation of $(\mathbf{x}, \mathbf{y})$ and (α, β), there exists a unique $(\mathbf{x}, \mathbf{y}) \in \mathcal{M}$ and unique reals α, β, and γ such that

$$\phi\left(\begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}, \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}\right) = \begin{pmatrix} \mathbf{v} \\ \mathbf{w} \end{pmatrix}. \tag{6.64}$$

The rest of the proof derives a closed-form expression of the inverse map ϕ^{-1}. First, using the Gram-Schmidt basis orthogonalization theorem, the family $(\mathbf{v}, \mathbf{w})$ can be expressed as

$$\begin{pmatrix}\mathbf{v}\\ \mathbf{w}\end{pmatrix} = \begin{pmatrix}c_1\mathbf{e}_1\\ c_2\mathbf{e}_1 + c_3\mathbf{e}_2\end{pmatrix} = \begin{pmatrix}c_1\mathbf{I}_n & 0\mathbf{I}_n\\ c_2\mathbf{I}_n & c_3\mathbf{I}_n\end{pmatrix}\begin{pmatrix}\mathbf{e}_1\\ \mathbf{e}_2\end{pmatrix} \tag{6.65}$$

with $c_1, c_3 > 0$ and $||\mathbf{e}_1||_2 = ||\mathbf{e}_2||_2 = 1$ and $\mathbf{e}_1^T\mathbf{e}_2 = 0$. The expression of these basis vectors and constants obtained from the Gram-Schmidt basis orthogonalization theorem are given by

$$\begin{aligned}c_1 &= ||\mathbf{v}||_2\\ \mathbf{e}_1 &= \frac{\mathbf{v}}{c_1}\\ c_2 &= \mathbf{w}^{\mathrm{T}}\mathbf{e}_1\\ c_3 &= ||\mathbf{w} - c_2\mathbf{e}_1||_2\\ \mathbf{e}_2 &= \frac{\mathbf{w} - c_2\mathbf{e}_1}{c_3}.\end{aligned}$$

Now, let $\mathbf{O} = \begin{pmatrix}\cos(\theta)\mathbf{I}_n & -\sin(\theta)\mathbf{I}_n\\ \cos(\varphi)\mathbf{I}_n & \sin(\varphi)\mathbf{I}_n\end{pmatrix}$ be an $2n \times 2n$ matrix for some angles θ and φ. Consider the non-isometric transformation of the basis vectors

$$\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix} = \mathbf{O}\begin{pmatrix}\mathbf{e}_1\\ \mathbf{e}_2\end{pmatrix}. \tag{6.66}$$

Notice that thanks to the structure of the matrix $\mathbf{O}$, the constraint $\mathbf{x}^{\mathrm{T}}\mathbf{x} = \mathbf{y}^{\mathrm{T}}\mathbf{y} = 1$. Indeed, we have $\mathbf{x}^{\mathrm{T}}\mathbf{x} = \cos(\theta)^2 + \sin(\theta)^2 = 1$. The constraint $\mathbf{x}^{\mathrm{T}}\mathbf{y} = c$ implies the following

$$\cos(\theta)\cos(\varphi) - \sin(\theta)\sin(\varphi) = c \Rightarrow \cos(\theta + \varphi) = c.$$

Now, combining (6.66) and (6.65) gives the following change of basis

$$\begin{pmatrix}\mathbf{v}\\ \mathbf{w}\end{pmatrix} = \begin{pmatrix}c_1\mathbf{I}_n & 0\mathbf{I}_n\\ c_2\mathbf{I}_n & c_3\mathbf{I}_n\end{pmatrix}\frac{1}{\det\{\mathbf{O}\}}\begin{pmatrix}\sin(\varphi)\mathbf{I}_n & \sin(\theta)\mathbf{I}_n\\ -\cos(\varphi)\mathbf{I}_n & \cos(\theta)\mathbf{I}_n\end{pmatrix}\begin{pmatrix}\mathbf{x}\\ \mathbf{y}\end{pmatrix}$$

Multiplying the above matrices an insisting that the component along $\mathbf{y}$ of the first vector matches the component along $\mathbf{x}$ of the second vector, i.e., the scalar γ is the same in (6.65), generate the following equality:

$$\begin{aligned}c_1\sin(\theta) &= c_2\sin(\varphi) - c_3\cos(\varphi)\\ c_1\frac{\sin(\theta)}{\cos(\varphi)} &= c_2\frac{\sin(\varphi)}{\cos(\varphi)} - c_3\\ c_1\frac{\sin(\theta)}{\cos(\varphi)} &= c_2\tan(\varphi) - c_3.\end{aligned} \tag{6.67}$$

Recall that $\theta = \cos^{-1}(c) - \varphi$. Therefore, we have

$$\sin(\theta) = \sin\Big(\cos^{-1}(c) - \varphi\Big) = \sqrt{1 - c^2}\cos\big(\varphi\big) - c\sin\big(\varphi\big). \tag{6.68}$$

Substituting the previous equality in (6.67) and rearranging the terms gives $c_1(\sqrt{1 - c^2} - c\tan\big(\varphi\big)) = c_2\tan\big(\varphi\big) - c_3$ and allows to conclude that

$$\tan\big(\varphi\big) = \frac{c_1\sqrt{1 - c^2} + c_3}{c_2 + c_1 c}. \tag{6.69}$$

The scalars α, β, and γ can be obtained by expanding (6.66). However, it is omitted herein as the retraction is only interested in the first component of the inverse map ϕ^{-1}.

Finally, the retraction is obtained by applying the above algorithm to the vector $(\mathbf{x} + \xi_{\mathbf{x}}, \mathbf{y} + \eta_{\mathbf{y}})$. In other words, the retraction is given by the following expression:

$$\mathsf{R}_{(\mathbf{x},\mathbf{y})}(\xi_{\mathbf{x}}, \eta_{\mathbf{y}}) = \pi_1\left(\phi^{-1}\left(\begin{pmatrix}\mathbf{x} + \xi_{\mathbf{x}}\\ \mathbf{y} + \eta_{\mathbf{y}}\end{pmatrix}\right)\right). \tag{6.70}$$

■

Remark 6.1 *Note that the arc-tangent function* $\tan^{-1}$ *in Theorem 6.3 is defined from* $[0, \pi)$ *instead of the usual* $[-\pi/2, \pi/2)$ *in order to obtain a positive scalar* α*, i.e., a positive sine.*

3-D GNSS Attitude Determination Using Riemannian Optimization

This part describes the proposed method for 3-D GNSS attitude determination using the above derived first and second-order Riemannian optimization algorithms. First, the ambiguity resolution method presented in (6.21) and (6.22) is applied to resolve the integer ambiguities. It should be noted that the antenna configuration shown in Figure 6.7 is utilized instead of the one in Figure 6.6. The estimations of the phase differences $\hat{\boldsymbol{\phi}}_{12}$ and $\hat{\boldsymbol{\phi}}_{14}$ are recovered. These phase differences can be used directly to obtain an estimate of the desired attitude. However, as pointed in [168], the unambiguous phase estimates usually suffer in terms of accuracy due to the accumulation of phase error. In addition, it is not guaranteed that the correct integer value is recovered by (6.22), i.e., the ambiguity resolution may actually fail. Therefore, these factors dictate that some processing should be performed before these phase estimates can be used for attitude determination.

Since the processes of ambiguity resolution for two baselines are independent, the geometry information of the antenna configuration is not respected. The prior knowledge of the angle between the two baselines can act as a constraint to judge the resolved integer ambiguities. If the angle condition isn't satisfied, a correction is performed to the resulting integer ambiguities by changing them to the closest integer until reaching a reasonable result. For a short baseline case, the bias of wrong integer ambiguity is usually very small.

Now, we are in a position to precisely estimate the pointing vectors using Riemannian Optimization (RieOpt method) developed in Chap. 2. The optimization problem in (6.29) is solved by the proposed RieOpt method initializing $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ with the LS solutions $\acute{\mathbf{x}}$ and $\acute{\mathbf{y}}$ as computed in (6.25). As mentioned before, the constraints in (6.29) strictly integrate the antennas geometry and baseline lengths into the cost function. Thus, as the optimization problem is ensured to converge to a critical point of the objective function, it is guaranteed that the geometry and the angles are satisfied.

In order to further improve the results, the phase difference between the longest baselines should be exploited. This can be done by applying the correction operation as in (6.26), and the step of integer ambiguity check and correction can be repeated again. Finally, the RieOpt method is carried out again for the largest baselines to refine the results of attitude determination. The integral process of the proposed method can be found in Algorithm 6.2.

Algorithm 6.2 Algorithm for 3-D GNSS attitude determination.

```
Recover $\hat{\boldsymbol{\phi}}_{12}$ and $\hat{\boldsymbol{\phi}}_{14}$ independently using (6.21)-(6.26).
Estimate the pointing vectors $\acute{\mathbf{x}}$ and $\acute{\mathbf{y}}$ using $\hat{\boldsymbol{\phi}}_{12}$ and $\hat{\boldsymbol{\phi}}_{14}$ by the least-squares solution (12).
while $|\acute{\mathbf{x}}^{\mathrm{T}}\acute{\mathbf{y}} - \cos(\Theta)| > \epsilon$ do
    Adjust the integer ambiguities to the closest integer.
    Re-compute $\hat{\boldsymbol{\phi}}_{12}$ and $\hat{\boldsymbol{\phi}}_{14}$ using the adjusted integer ambiguities.
    Re-compute $\acute{\mathbf{x}}$ and $\acute{\mathbf{y}}$ based on (12).
end while
Estimate the pointing vectors $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ using the RieOpt method initialized using $\acute{\mathbf{x}}$ and $\acute{\mathbf{y}}$.
Compute $\hat{\boldsymbol{\phi}}_{13}$ and $\hat{\boldsymbol{\phi}}_{15}$ from $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ using (13).
Re-compute $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ using $\hat{\boldsymbol{\phi}}_{13}$ and $\hat{\boldsymbol{\phi}}_{15}$ through the RieOpt method described in Chap. 2.
```

The proposed Riemannian steepest-descent method in Algorithm 6.2 requires the computation of the Riemannian gradient and the retraction at each step. As shown previously, the Riemannian gradient is obtained by simply projecting the Euclidean gradient. Given that the derived orthogonal projection only involves inner products, its complexity is linear in the dimension of the system. Likewise, from the expression of the retraction, the most expensive step is the matrix multiplication, which is quadratic in n. However, given the tri-diagonal structure of the matrix, the multiplication can be done in linear time, which results in a retraction with linear complexity. Finally, it can be concluded that the per iteration complexity of the proposed Riemannian steepest-descent method is linear in the dimension of the problem. Furthermore, thanks to the compactness of the manifold, first-order Riemannian methods are shown to exhibit a linear convergence rate.

The complexity analysis of the proposed Riemannian Newton's method follows similar steps as for the steepest-descent. Indeed, it can easily be seen that the most complex step is to compute the Hessian and solve for the search direction. From the expression of the Hessian in Proposition 6.1, we can easily see that it can be computed in $O(n)$ steps. However, solving for the search direction $\xi_{\mathbf{x}}$ that satisfies $\mathsf{hess}\, \mathsf{f}(\mathbf{x})[\xi_{\mathbf{x}}] = -\mathsf{grad}\, \mathsf{f}(\mathbf{x})$ requires solving a system of n linear equations which can be done is n^3. However, the use of iterative methods can reduce the complexity to n^2 resulting in an algorithm with quadratic complexity. Thanks to the use of second-order derivatives, the proposed Newton's method exhibits a convergence rate that is at least quadratic.

Simulation Setup and Results

This subsection presents the simulation results in order to evaluate the performance of the proposed approach under a noise-controlled environment by plotting the performance of the proposed method against benchmarks methods in the literature, namely, the least-squares, LAMBDA method [160], and generic non-convex solvers. The LAMBDA method is the standard method for attitude determination and has found success in different attitude scenarios. The LS method resolves the integer ambiguity and estimates the attitude using the (6.21)-(6.26) for each baseline separately. A detailed description of this method can be found in [168]. The proposed RieOpt method applies the procedure summarized in Algorithm 6.2. This method solves the optimization problem to refine the results obtained from the procedure given by (6.21)-(6.26), and obtain a final attitude estimate. While the LS and the proposed approach utilizes only carrier phase, the LAMBDA [160] and MC-LAMBDA [179] methods take advantage of both the carrier phase and pseudo-range. However, the LAMBDA method does not utilize the a-priori knowledge of the antenna array geometry. Finally, similar to the proposed methods, the MC-LAMBDA method leverages all the available geometrical information.

Table 6.1: Success rates over many GPS weeks.

Date	GPS week	LS		RieOpt		LAMBDA	
		x	y	x	y	x	y
03-Sep-2017	941	0.960	0.960	0.961	0.962	0.973	0.974
01-Oct-2017	945	0.982	0.984	0.984	0.985	0.998	0.998
19-Nov-2017	952	0.987	0.988	0.988	0.989	0.999	0.988
28-Jan-2018	962	0.988	0.988	0.988	0.988	0.999	0.999
04-Feb-2018	963	0.985	0.987	0.984	0.987	0.999	0.999
01-Apr-2018	971	0.981	0.982	0.983	0.984	0.998	0.998
08-Apr-2018	972	0.980	0.981	0.983	0.984	0.998	0.998
22-Apr-2018	974	0.969	0.970	0.972	0.973	0.991	0.992
29-Apr-2018	975	0.982	0.983	0.984	0.985	0.999	0.999
06-May-2018	976	0.983	0.984	0.985	0.985	0.998	0.998

Table 6.2: Root mean square error over different GPS weeks.

Date	GPS week	LS		RieOpt		LAMBDA	
		x	y	x	y	x	y
03-Sep-2017	941	0.465	0.458	0.338	0.331	0.651	0.637
01-Oct-2017	945	0.338	0.367	0.268	0.280	0.478	0.519
19-Nov-2017	952	0.521	0.536	0.359	0.382	0.734	0.755
28-Jan-2018	962	0.734	0.746	0.489	0.498	1.035	1.052
04-Feb-2018	963	0.555	0.637	0.416	0.541	0.783	0.900
01-Apr-2018	971	0.349	0.343	0.265	0.263	0.491	0.483
08-Apr-2018	972	0.304	0.307	0.251	0.252	0.428	0.433
22-Apr-2018	974	0.301	0.317	0.253	0.256	0.424	0.447
29-Apr-2018	975	0.330	0.331	0.271	0.271	0.465	0.467
06-May-2018	976	0.379	0.394	0.284	0.287	0.532	0.555

The simulations are implemented using MATLAB and the libraries of the visual simulation software [180]. The basic parameters are produced based on the libraries of this software using the receiver location or antenna baseline and real GPS constellation information. Furthermore, the MATLAB version 3.0 of the LAMBDA software [181] is used.

The presence of trees, buildings and other high rising structures, natural or artificial, around the receivers may hamper the view of the satellites. GNSS observations below 15 degrees are usually disturbed by multipath and other problems, e.g., cycle slips, and low signal-to-noise ratio [167]. The elevation mask allows us to set an elevation angle such that we can discard the data below that angle. This parameter has a great influence on dilution of precision (DoP), so it needs to be chosen carefully. In our simulations, an elevation mask of 15 degrees is applied.

The proposed method can be easily modified and extended to multi-epoch and multi-frequency cases. However, in our simulations, only the challenging single-epoch, single-frequency case is considered using the GPS L1 frequency, which has a wavelength equal to 19cm approximately. The antenna configuration is similar to Figure 6.7 with $\triangle = \triangle' = 8$ cm and different values of the angle Θ between the two baselines are considered. According to this setup, both conditions (6.20) and (6.27) are satisfied. It should be noted that this configuration is not required for the LAMBDA and MC-LAMBDA methods, which are able to perform ambiguity resolution using any arbitrary configuration. As aforementioned, this special configuration is used as a means to simplify the ambiguity resolution process. Besides, the proposed Riemannian manifold optimization can be paired with any ambiguity resolution method, as will be demonstrated in this section.

To demonstrate the performance of the proposed method in multiple possible scenarios, simulations are carried out over different GPS weeks, noise levels and number of satellites. For each simulation setup, the tests are repeated 10^5 times with random Gaussian noise and random baseline pointing vectors. For the carrier phase observables, the Gaussian noise has zero mean and standard deviation σ_ϕ, ranging from 1 mm to 7 mm. Unlike other non-convex methods, the proposed Riemannian method is guaranteed to always reach an extreme point of the problem. Furthermore, for moderate to high noise variance, non-convex solvers return different results due to the existence of multiple local minima. Typical GPS phase measurements have phase standard deviation in the range of $1 - 3$ mm [182].For GPS, the pseudo-range and phase variance ratio is around 10^4 [155], i.e., $\dfrac{\sigma_p^2}{\sigma_\phi^2} = 10^4$. For pseudo-range measurements, according to the above radio, the Gaussian noises with zero-mean and standard deviation σ_p are used.

To test the performance of the proposed approach, we evaluate the success rate and root mean square error (RMSE) of the baseline pointing direction. Success rate is defined as the percentage of occurrences that the integer vectors are correctly fixed over 10^5 simulation trials. The baseline pointing vector error refers to the deviation of the resulting direction vector from the true pointing direction. Finally, the RMSE is evaluated only for the configurations resulting in a successful estimation.

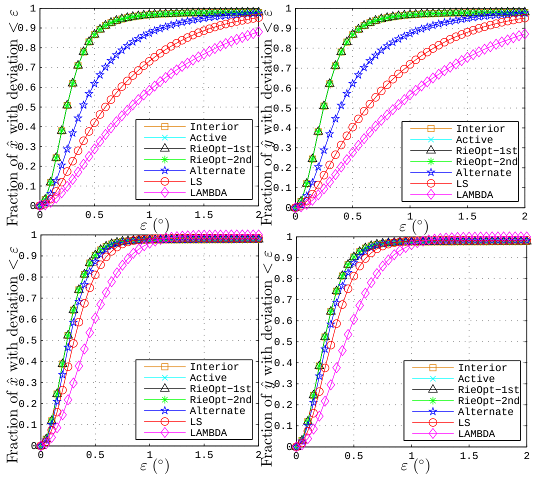


Figure 6.8: Fraction of estimates with error less than ϵ over different GPS weeks.

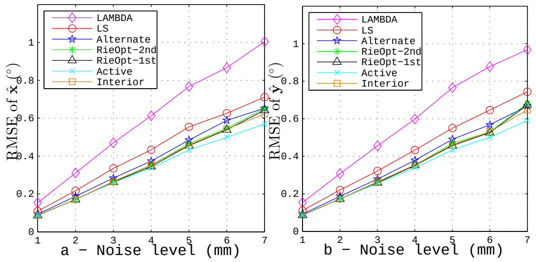


Figure 6.9: Root mean square error versus noise levels (29/4/2018).

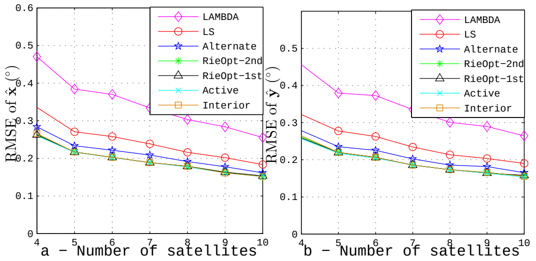


Figure 6.10: Root mean square error versus the number of satellites (29/4/2018).

Table 6.1 shows the success rates of the proposed and the two benchmark approaches for 10 different GPS weeks on specific days and time $00 : 00$. In this result set, the same setup is used with $d_{12} = d_{14} = 45$ cm, $\triangle = \triangle' = 8$ cm, $\sigma_\phi = 3$ mm, $\Theta = 90^\circ$ and number of satellites $m = 4$. Such a small number of satellites is meant to provide a challenging scenario to compare the performance of three methods. Table 6.1 indicates that all the three methods have very high success rates. The RieOpt method and LS method have similar success rates since the RieOpt method

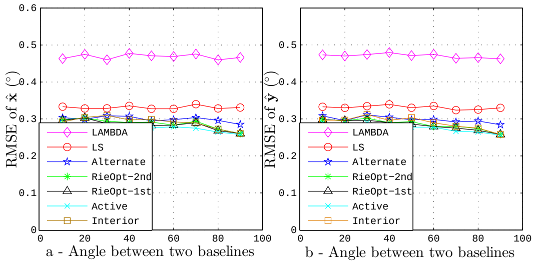


Figure 6.11: Root mean square error versus the angle (29/4/2018).

is based on the unwrapped phase obtained from the LS method. However, the success rate of the LAMBDA method is a little higher than the other two methods, which may come from the fact that the LAMBDA method also takes advantage of pseudo-range data. Even though the success rates are slightly different, all the three methods offer excellent performance in satellite-deprived environments.

Table 6.2 lists the RMSE of the three methods over the 10 GPS weeks of Table 6.1. It is clear that the RieOpt method provides the best accuracy in almost all simulated cases except few instances wherein the Active set method provides better results. The RieOpt method rigorously integrates the antenna geometry and baseline lengths to its objective function so that it produces more accurate results than the other methods. However, it is worth mentioning that the LS method also outperforms the LAMBDA method.

Figure 6.8 shows the fraction of simulation trials with RMSE below different values on two different dates, January 28, 2018 and April 29, 2018. These plots depict the success rates and attitude estimation accuracy of the seven methods in a different way. These four plots show that the RieOpt method can enhance the accuracy of the resulting pointing vectors while offering similar success rates to those of the LS. Comparing the plots of these two GPS weeks, we can find out that the improvement is more evident for the case in which the LS method and LAMBDA method have less accurate results.

To test the effect of carrier phase noise on the proposed method, a set of simulations is carried out for GPS week 975. The standard deviation of carrier phase, σ_ϕ, varies from 1 mm to 7 mm, and the standard deviation of pseudo-range adjusted according to the ratio $\sigma_p^2/\sigma_\phi^2 = 10^4$. All the other setting parameters are kept unchanged. Figure 6.9 plots the corresponding RMSE of the three methods under different noise levels. Although the errors of all the methods increase as the noises increase, the RieOpt method consistently offers the best performance in almost all scenarios. The

LS method outperforms the LAMBDA method over the different scenarios. Besides, the larger the noise is, the more obvious the accuracy improvement of the RieOpt method is. Figure 6.10 shows the RMSE of all methods when different numbers of satellites are used. As expected, the accuracy of all the methods improves as the the number of satellites increases. The results prove that the RieOpt method provides the best performance in almost all situations regardless of the number of satellites. The proposed methods outperform the LAMBDA method in terms of the RMSE, which is mainly attributed to the incorporation of the geometry in the optimization process. The success rate of the proposed method is slightly lower than that of the LAMBDA method. An explanation of this is that the difference operation in (6.21) exacerbates the effect of the phase error, which affects the final success rate.

Table 6.3: Performance of different GNSS attitude determination algorithms.

Method	Time(s)	RMSE (°) x	RMSE (°) y
LAMBDA	0.0038	1.0216	1.1002
MC-LAMBDA	0.0120	0.2592	0.2615
Rie-1st	0.0094	0.2596	0.2591
Rie-2nd	0.0255	0.2592	0.2606
LS	0.0038	0.2855	0.2879
Alternate	0.2224	0.2699	0.2762
Interior	0.0468	0.2615	0.2691
Active	0.0313	0.2696	0.2701

While the above mentioned simulations assume orthogonal baselines, Figure 6.11 attest the performance of the proposed algorithm using a more-general configuration, i.e., with different angle values between two baselines. From the numerical results, it appears that the angle between two baselines does not show any visible effect on the results. Finally, considering all the aforementioned simulations, it can be concluded that the RieOpt approach is able to significantly improve the accuracy of attitude estimation over that offered by the LS, the LAMBDA methods, and generic non-convex solvers. Finally, as stated earlier, the proposed Riemannian methods can be combined with the integer ambiguity resolution from the MC-LAMBDA method. Table 6.3 shows the performance and the runtime for the LAMBDA, MC-LAMBDA, LS, generic non-convex solvers, and the proposed algorithms. From the table, it can easily be seen that our method provides similar performance to that of the MC-LAMBDA but for a lower complexity. Indeed, the table shows that our method is on average 25% faster than MC-LAMBDA. This further emphasizes the clever exploitation of the geometry of the problem by the proposed Riemannian methods.

Chapter 7

CONCLUSION

This thesis exploits a recent optimization technique known as Riemannian optimization for solving convex and non-convex signal processing and machine learning problems. While the complexity of traditional optimization methods, such as interior-point methods and semi-definite programs, increases with the number of constraints of the problem, Riemannian optimization cleverly reformulates the problem on the feasible space which results in a reduction of the embedded dimension and thus efficient optimization algorithms. The required ingredients for designing first-order Riemannian optimization schemes are derived for multiple contemporary convex and non-convex examples in signal processing and machine learning, such as community detection, graph-based clustering, phase retrieval, and indoor and outdoor location determination. Even though the resulting Riemannian algorithms are not universal but rather problem dependent, the broad set of available implementations in the literature for different optimization algorithms is turning Riemannian optimization into a toolbox that can be used by the scientific community at large regardless of their knowledge on differential geometry and Riemannian manifolds.

BIBLIOGRAPHY

[1] J. A. De Loera, R. Hemmecke, K. Matthiasand, et al. *Algebraic and geometric ideas in the theory of discrete optimization*. Vol. 14. SIAM, 2013.

[2] A. Bagirov, N. Karmitsa, and M. M. Mkel. *Introduction to Nonsmooth Optimization: Theory, Practice and Software*. Springer Publishing Company, Incorporated, 2014. ISBN: 3319081136.

[3] A. R. Conn, K. Scheinberg, and L. N. Vicente. *Introduction to derivative-free optimization*. Vol. 8. Siam, 2009.

[4] G. B. Dantzig. *Linear programming and extensions*. Vol. 48. Princeton university press, 1998.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004. ISBN: 0521833787.

[6] H. D. Sherali and W. P. Adams. *A reformulation-linearization technique for solving discrete and continuous nonconvex problems*. Vol. 31. Springer Science & Business Media, 2013.

[7] S. Bubeck et al. "Convex optimization: Algorithms and complexity". In: *Foundations and Trends® in Machine Learning* 8.3-4 (2015), pp. 231–357.

[8] P. Jain, P. Kar, et al. "Non-convex optimization for machine learning". In: *Foundations and Trends® in Machine Learning* 10.3-4 (2017), pp. 142–336.

[9] S. K. Mishra. *Topics in Nonconvex Optimization*. Springer, 2011.

[10] R. G. Strongin and Y. D. Sergeyev. *Global optimization with non-convex constraints: Sequential and parallel algorithms*. Vol. 45. Springer Science & Business Media, 2013.

[11] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006, 2006.

[12] D. P. Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[13] I. Csiszár. "Information geometry and alternating minimization procedures". In: *Statistics and decisions* 1 (1984), pp. 205–237.

[14] U. Niesen, D. Shah, and G. Wornell. "Adaptive alternating minimization algorithms". In: *Proc. of the IEEE International Symposium on Information Theory (ISIT' 2017), Aachen, Germany*. 2007, pp. 1641–1645.

[15] C. L. Byrne. "Alternating minimization and alternating projection algorithms: A tutorial". In: *Sciences New York* (2011), pp. 1–41.

[16] W. Ring and B. Wirth. “Optimization Methods on Riemannian Manifolds and Their Application to Shape Space”. In: *SIAM Journal on Optimization* 22.2 (2012), pp. 596–627.

[17] D. G. Luenberger. “The Gradient Projection Method Along Geodesics”. In: *Management Science* 18.11 (1972), pp. 620–631.

[18] D. Gabay. “Minimizing a differentiable function over a differential manifold”. In: *Journal of Optimization Theory and Applications* 37.2 (1982), pp. 177–219.

[19] S. T. Smith. *Geometric Optimization Methods for Adaptive Filtering*. UMI Order No. GAX93-31032. Cambridge, MA, USA: Harvard University, 1993.

[20] C. Udriste. *Convex functions and optimization methods on Riemannian manifolds*. Vol. 297. Springer Science & Business Media, 1994.

[21] Y. Yang. “Globally Convergent Optimization Algorithms on Riemannian Manifolds: Uniform Framework for Unconstrained and Constrained Optimization”. In: *Journal of Optimization Theory and Applications* 132.2 (2007), pp. 245–265.

[22] H. Ji. *Optimization approaches on smooth manifolds*. Australian National University, 2007.

[23] P. A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ: Princeton University Press, 2008.

[24] K. Huper and J. Trumpf. “Newton-like methods for numerical optimization on manifolds”. In: *Proc. of the 38th Asilomar Conference on Signals, Systems and Computers, 2004, Pacific Grove, CA, USA*. Vol. 1. Nov. 2004, 136–139 Vol.1.

[25] C. G. Baker, P. A. Absil, and K. A. Gallivan. “An implicit trust-region method on Riemannian manifolds”. In: *IMA J. Numer. Anal.* 28.4 (2008), pp. 665–689. DOI: doi:10.1093/imanum/drn029.

[26] C.G. Baker. *Riemannian Manifold Trust-region Methods with Applications to Eigenproblems*. The Florida State University, 2008.

[27] P. A. Absil, C. G. Baker, and K. A. Gallivan. “Trust-region methods on Riemannian manifolds”. In: *Foundations of Computational Mathematics* 7.3 (2007), pp. 303–330.

[28] P. A. Absil, C. G. Baker, and K. A. Gallivan. “Trust-region methods on Riemannian manifolds”. In: *Proc. of the 16th International Symposium on Mathematical Theory of Networks and Systems (MTNS’ 04), Leuven, Belgium*. 2004.

[29] U. Shalit, D. Weinshall, and G. Chechik. “Online Learning in the Embedded Manifold of Low-rank Matrices”. In: *Journal of Machine Learning Research* 13.1 (Feb. 2012), pp. 429–458. ISSN: 1532-4435.

[30] R. Inokuchi and S. Miyamoto. “c-Means Clustering on the Multinomial Manifold”. In: *Proc. of the 4th International Conference on Modeling Decisions for Artificial Intelligence (MDAI ’07), Kitakyushu, Japan*. 2007, pp. 261–268.

[31] J. Lafferty and G. Lebanon. “Diffusion Kernels on Statistical Manifolds”. In: *Journal of Machine Learning Research* 6 (Dec. 2005), pp. 129–163.

[32] Y. Sun et al. “Heterogeneous Tensor Decomposition for Clustering via Manifold Optimization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.3 (Mar. 2016), pp. 476–489.

[33] N. Boumal and P. A. Absil. “RTRMC: A Riemannian trust-region method for low-rank matrix completion”. In: *Proc. of the Advances in neural information processing systems (NIPS’ 2011), Granada, Spain*. 2011, pp. 406–414.

[34] B. Vandereycken. “Low-rank matrix completion by Riemannian optimization”. In: *SIAM Journal on Optimization* 23.2 (2013), pp. 1214–1236.

[35] L. Cambier and P. A. Absil. “Robust low-rank matrix completion via Riemannian optimization”. In: *SIAM Journal on Scientific Computing* 38.5 (2015), S440–S460. doi: 10.1137/15M1025153.

[36] A. Douik and B. Hassibi. “An Improved Initialization for Low-Rank Matrix Completion Based on Rank-1 Updates”. In: *Proc. of the IEEE International Conf. on Acoustics Speech and Signal Processing (ICASSP’ 2018), Calgary, AL, Canada*. June 2018, pp. 1–5. doi: 10.1109/ICASSP.2018.8461826.

[37] N. Boumal et al. “Manopt, a Matlab Toolbox for Optimization on Manifolds”. In: *Journal of Machine Learning Research* 15 (2014), pp. 1455–1459. url: http://www.manopt.org.

[38] A. K. Jain, M. N Murty, and P. J. Flynn. “Data clustering: A review”. In: *ACM computing surveys (CSUR)* 31.3 (1999), pp. 264–323.

[39] R. K. Vinayak and B. Hassibi. “Similarity clustering in the presence of outliers: Exact recovery via convex program”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2016), Barcelona, Spain*. July 2016, pp. 91–95.

[40] H. Xu, C. Caramanis, and S. Sanghavi. “Robust PCA via outlier pursuit”. In: *Proc. of the Advances in Neural Information Processing Systems (NIPS’ 2010), Vancouver, BC, Canada*. 2010, pp. 2496–2504.

[41] Y. Chen et al. “Clustering partially observed graphs via convex optimization”. In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 2213–2238.

[42] J. R. Fienup. “Phase retrieval algorithms: A personal tour”. In: *Applied optics* 52.1 (2013), pp. 45–56.

[43] M. Soltanolkotabi. “Structured signal recovery from quadratic measurements: Breaking sample complexity barriers via nonconvex optimization”. In: *IEEE Transactions on Information Theory* 65.4 (2019), pp. 2374–2400.

[44] A. Douik, F. Salehi, and B. Hassibi. “A Novel Riemannian Optimization Approach and Algorithm for Solving the Phase Retrieval Problem”. In: *Proc. of the 53rd Asilomar Conference on Signals, Systems, and Computers (Asilomar’ 2019), Asilomar, CA, USA*. Vol. 1. 1. Nov. 2019, pp. 1962–1966. DOI: 10.1109/IEEECONF44664.2019.9049040.

[45] F. Ijaz et al. “Indoor positioning: A review of indoor ultrasonic positioning systems”. In: *Proc. of the 15th International Conference on Advanced Communications Technology (ICACT’13), PyeongChang, Korea*. IEEE. 2013, pp. 1146–1150.

[46] S. J. Kim and B. K. Kim. “Dynamic ultrasonic hybrid localization system for indoor mobile robots”. In: *IEEE Transactions on Industrial Electronics* 60.10 (2012), pp. 4562–4573.

[47] R. Zhang, F. Höflinger, and L. Reindl. “TDOA-based localization using interacting multiple model estimator and ultrasonic transmitter/receiver”. In: *IEEE Transactions on Instrumentation and Measurement* 62.8 (2013), pp. 2205–2214.

[48] C. Ferrando, A. Perez, and R. S. Sanchez Pena. “Integer ambiguity resolution in GPS for spinning spacecrafts”. In: *IEEE Transactions on Aerospace and Electronic Systems* 35.4 (Oct. 1999), pp. 1219–1229.

[49] Y. Li et al. “On-the-fly GPS-based attitude determination using single- and double-differenced carrier phase measurements”. In: *GPS Solutions* 8.2 (July 2004), pp. 93–102.

[50] K. Chiang et al. “GPS-based attitude determination for a spinning rocket”. In: *IEEE Transactions on Aerospace and Electronic Systems* 50.4 (Oct. 2014), pp. 2654–2663.

[51] P. J. G. Teunissen, G. Giorgi, and P. J. Buist. “Testing of a new single-frequency GNSS carrier phase attitude determination method: land, ship and aircraft experiments”. In: *GPS Solutions* 15.1 (Jan. 2011), pp. 15–28.

[52] A. Douik et al. “Precise 3-D GNSS Attitude Determination Based on Riemannian Manifold Optimization Algorithms”. In: *IEEE Transactions on Signal Processing* 68.1 (Dec. 2020), pp. 284–299. DOI: 10.1109/TSP.2019.2959226.

[53] Z. Yang and E. Oja. “Clustering by Low-rank Doubly Stochastic Matrix Decomposition”. In: (2012), pp. 707–714.

[54] X. Wang, F. Nie, and H. Huang. “Structured Doubly Stochastic Matrix for Graph Based Clustering: Structured Doubly Stochastic Matrix”. In: *Proc. of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD’ 2016), San Francisco, CA, USA*. 2016, pp. 1245–1254. ISBN: 978-1-4503-4232-2.

[55] R. Zass and A. Shashua. “Doubly stochastic normalization for spectral clustering”. In: *Proc. of the Advances in Neural Information Processing Systems (NIPS’ 2006),Vancouver, B.C., Canada*. 2006, pp. 1569–1576.

[56] F. Salehi, E. Abbasi, and B. Hassibi. “A precise analysis of phasemax in phase retrieval”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2018), CO, USA*. IEEE. 2018, pp. 976–980.

[57] K. Jaganathan, Y. C. Eldar, and B. Hassibi. “Phase retrieval: An overview of recent developments”. In: *arXiv preprint arXiv:1510.07713* (2015).

[58] T. Bendory, R. Beinert, and Y. C. Eldar. “Fourier phase retrieval: Uniqueness and algorithms”. In: Springer, 2017, pp. 55–91.

[59] K. Huang, Y. C. Eldar, and N. D. Sidiropoulos. “Phase retrieval from 1D Fourier measurements: Convexity, uniqueness, and algorithms”. In: *IEEE Transactions on Signal Processing* 64.23 (2016), pp. 6105–6117.

[60] P. Davidson and R. Piché. “A survey of selected indoor positioning methods for smartphones”. In: *IEEE Communications Surveys & Tutorials* 19.2 (2016), pp. 1347–1370.

[61] Y. Gu, A. Lo, and I. Niemegeers. “A survey of indoor positioning systems for wireless personal networks”. In: *IEEE Communications surveys & tutorials* 11.1 (2009), pp. 13–32.

[62] G. Borriello. “Location systems for ubiquitous computing”. In: *IEEE computer* (2000), pp. 57–66.

[63] K. Whitehouse, C. Karlof, and D. Culler. “A practical evaluation of radio signal strength for ranging-based localization”. In: *ACM Sigmobile Mobile Computing and Communications Review* 11.1 (2007), pp. 41–52.

[64] C. Yüzbaşioğlu and B. Barshan. “Improved range estimation using simple infrared sensors without prior knowledge of surface characteristics”. In: *Measurement Science and Technology* 16.7 (2005), p. 1395.

[65] M. C. Amann et al. “Laser ranging: A critical review of unusual techniques for distance measurement”. In: *Optical engineering* 40 (2001).

[66] R. B. Langley et al. “Dilution of Precision”. In: *GPS world* 10.5 (1999), pp. 52–59.

[67] G. Giorgi and P. J. G. Teunissen. “Low-complexity instantaneous ambiguity resolution with the affine-constrained GNSS attitude model”. In: *IEEE Transactions on Aerospace and Electronic Systems* 49.3 (2013), pp. 1745–1759.

[68] G. Giorgi and P. J. G. Teunissen. “GNSS carrier phase-based attitude determination”. In: InTech, 2012, pp. 193–220.

[69] M. Berger and B. Gostiaux. *Differential geometry: manifolds, curves, and surfaces*. Graduate texts in mathematics. Springer-Verlag, 1988.

[70] J. Lee. *Introduction to Topological Manifolds*. Graduate Texts in Mathematics. Springer New York, 2010.

[71] P. Petersen. *Riemannian Geometry*. Graduate texts in mathematics. Springer, 1998.

[72] T. T. Yao et al. “A Riemannian Fletcher-Reeves Conjugate Gradient Method for Doubly Stochastic Inverse Eigenvalue Problems”. In: *SIAM Journal on Matrix Analysis and Applications* 37.1 (2016), pp. 215–234.

[73] L. M. Adams, J. L. Nazareth, et al. *Linear and nonlinear conjugate gradient-related methods*. Vol. 85. Siam, 1996.

[74] A. Douik and B. Hassibi. “A Riemannian Approach for Graph-Based Clustering by Doubly Stochastic Matrices”. In: *Proc. of the IEEE Statistical Signal Processing Workshop (SSP’ 2018), Freiburg, Germany*. Vol. 1. 1. June 2018, pp. 806–810. DOI: `10.1109/SSP.2018.8450685`.

[75] A. Douik and B. Hassibi. “Manifold Optimization Over the Set of Doubly Stochastic Matrices: A Second-Order Geometry”. In: *IEEE Transactions on Signal Processing* 67.22 (Nov. 2019), pp. 5761–5774. DOI: `10.1109/TSP.2019.2946024`.

[76] D. J. Hartfiel and J. W. Spellmann. “A role for doubly stochastic matrices in graph theory”. In: *Proceedings of the American Mathematical Society* 36.2 (1972), pp. 389–394.

[77] K. Shanmugam, A. G. Dimakis, and M. Langberg. “Local graph coloring and index coding”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2013), Istanbul, Turkey*. IEEE. 2013, pp. 1152–1156.

[78] N. A. Ruhi and P. Bogdan. “Multiscale modeling of biological communication”. In: *Proc. of the IEEE International Conference on Communications (ICC’ 2015), London, UK*. Vol. 1. 1. June 2015, pp. 1140–1145.

[79] N. A. Ruhi, H. J. Ahn, and B. Hassibi. “Analysis of exact and approximated epidemic models over complex networks”. In: *arXiv preprint arXiv:1609.09565* (2016).

[80] R. A. Brualdi. “Some applications of doubly stochastic matrices”. In: *Linear Algebra and its Applications* 107 (1988), pp. 77–100.

[81] M. Mehlum. “Doubly Stochastic Matrices and the Assignment Problem”. MA thesis. University of Oslo, 2012.

[82] S. Amari and H. Nagaoka. *Methods of Information Geometry*. Translations of mathematical monographs. American Mathematical Society, 2007.

[83] G. Hurlbert. “A short proof of the Birkhoff-von Neumann Theorem”. In: *preprint (unpublished)* (2008).

[84] R. Sinkhorn. “A relationship between arbitrary positive matrices and doubly stochastic matrices”. In: *The annals of mathematical statistics* 35.2 (1964), pp. 876–879.

[85] R. Sinkhorn and P. Knopp. “Concerning nonnegative matrices and doubly stochastic matrices”. In: *Pacific Journal of Mathematics* 21.2 (1967), pp. 343–348.

[86] J. Csima and B. N. Datta. “The DAD theorem for symmetric non-negative matrices”. In: *Journal of Combinatorial Theory, Series A* 12.1 (1972), pp. 147–152. ISSN: 0097-3165.

[87] M. Idel. “A review of matrix scaling and Sinkhorn’s normal form for matrices and positive maps”. In: *ArXiv preprint* (2016). URL: `https://arxiv.org/abs/1609.06349`.

[88] A. Khosla et al. “Novel dataset for fine-grained image categorization: Stanford dogs”. In: *Proc. of the CVPR Workshop on Fine-Grained Visual Categorization (FGVC’ 2011), Honolulu, HI, USA*. Vol. 2. 2011, p. 1.

[89] A. Douik and B. Hassibi. “Low-Rank Riemannian Optimization on Positive Semidefinite Stochastic Matrices with Applications to Graph Clustering”. In: *Proc. of the International Conference on Machine Learning (ICML’ 2018), Stockholm, Sweden*. Vol. 80. July 2018, pp. 1299–1308. URL: `proceedings.mlr.press/v80/douik18a.html`.

[90] G. A. Watson. “Characterization of the subdifferential of some matrix norms”. In: *Linear algebra and its applications* 170 (1992), pp. 33–45.

[91] A. Douik and B. Hassibi. “Low-Rank Riemannian Optimization for Graph-Based Clustering Applications”. In: *Submitted to IEEE Transactions on Pattern Analysis and Machine Intelligence* P.P (Dec. 2020), PP.

[92] A. Douik and B. Hassibi. “Non-Negative Matrix Factorization via Low-Rank Stochastic Manifold Optimization”. In: *Proc. of the IEEE International Symposium on Information Theory (ISIT’ 2019), Paris, France*. Vol. 1. 1. June 2019, pp. 497–501. DOI: `10.1109/ISIT.2019.8849441`.

[93] Z. Yang and E. Oja. “Unified Development of Multiplicative Algorithms for Linear and Quadratic Nonnegative Matrix Factorization”. In: *IEEE Transactions on Neural Networks* 22.12 (Dec. 2011), pp. 1878–1891.

[94] C. H. Q. Ding, T. Li, and M. I. Jordan. “Convex and semi-nonnegative matrix factorizations”. In: *IEEE transactions on pattern analysis and machine intelligence* 32.1 (2010), pp. 45–55.

[95] V. Chandrasekaran et al. “The convex geometry of linear inverse problems”. In: *Foundations of Computational mathematics* 12.6 (2012), pp. 805–849.

[96] R. Arora et al. “Clustering by left-stochastic matrix factorization”. In: *Proc. of the 28th International Conference on Machine Learning (ICML’ 2011), Bellevue, WA, USA*. 2011, pp. 761–768.

[97] S. Homer and M. Peinado. “Design and performance of parallel and distributed approximation algorithms for maxcut”. In: *Journal of Parallel and Distributed Computing* 46.1 (1997), pp. 48–61.

[98] C. Helmberg and F. Rendl. “A spectral bundle method for semidefinite programming”. In: *SIAM Journal on Optimization* 10.3 (2000), pp. 673–696.

[99] S. Burer and R. D. C. Monteiro. “A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization”. In: *Mathematical Programming* 95.2 (2003), pp. 329–357.

[100] I. Grubišić and R. Pietersz. “Efficient rank reduction of correlation matrices”. In: *Linear algebra and its applications* 422.2-3 (2007), pp. 629–653.

[101] S. Bonnabel and R. Sepulchre. “Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank”. In: *SIAM Journal on Matrix Analysis and Applications* 31.3 (2009), pp. 1055–1070.

[102] M. Journée et al. “Low-rank optimization on the cone of positive semidefinite matrices”. In: *SIAM Journal on Optimization* 20.5 (2010), pp. 2327–2351.

[103] J. Kyparisis. “On uniqueness of Kuhn-Tucker multipliers in nonlinear programming”. In: *Mathematical Programming* 32.2 (1985), pp. 242–246.

[104] C. M. Bishop. *Neural Networks for Pattern Recognition*. New York, NY, USA: Oxford University Press, Inc., 1995. ISBN: 0198538642.

[105] M. Buhrmester, T. Kwang, and S. D. Gosling. “Amazon’s Mechanical Turk: A new source of inexpensive, yet high-quality, data?” In: *Perspectives on psychological science* 6.1 (2011), pp. 3–5.

[106] M. Grant and S. Boyd. *CVX: Matlab Software for Disciplined Convex programming, version 2.1*. `http://cvxr.com/cvx`. Mar. 2014.

[107] Z. Lin, M. Chen, and Y. Ma. “The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices”. In: *arXiv preprint arXiv:1009.5055* (2010).

[108] Marina Meilă. “Comparing clusterings by the variation of information”. In: Springer, 2003, pp. 173–187.

[109] Z. Yang, J. Corander, and E. Oja. “Low-rank doubly stochastic matrix decomposition for cluster analysis”. In: *Journal of Machine Learning Research* 17.187 (2016), pp. 1–25.

[110] S. Siddiqi, B. Boots, and G. Gordon. “Reduced-rank hidden Markov models”. In: *Proc. of the Thirteenth International Conference on Artificial Intelligence and Statistics, (AISTATS’ 10), Sardinia, Italy*. 2010, pp. 741–748.

[111] N. Srebro A. Bijral. “On Doubly Stochastic Graph Optimization”. In: *Proc. of the 22nd Annual Conference on Neural Information Processing Systems (NIPS’ 2009), Vancouver, B.C., Canada*. 2009, pp. 1569–1576.

[112] N. Boumal, V. Voroninski, and A. Bandeira. “The non-convex Burer-Monteiro approach works on smooth semidefinite programs”. In: *Proc. of the Advances in Neural Information Processing Systems (NIPS’ 2016), Barcelona, Spain*. 2016, pp. 2757–2765.

[113] D. D. Lee and H. S. Seung. “Algorithms for non-negative matrix factorization”. In: *Proc. of the Advances in neural information processing systems (NIPS’ 2001), Vancouver, B.C., Canada*. 2001, pp. 556–562.

[114] N. Gillis. “The why and how of nonnegative matrix factorization”. In: *Regularization, Optimization, Kernels, and Support Vector Machines* 12.257 (2014).

[115] C. Adams. “Stochastic matrix factorization”. In: *Available at SSRN 2840852* (2016).

[116] B. Mishra et al. “Fixed-rank matrix factorizations and Riemannian low-rank optimization”. In: *Computational Statistics* 29.3-4 (2014), pp. 591–621.

[117] P. A. Absil, L. Amodei, and G. Meyer. “Two Newton methods on the manifold of fixed-rank matrices endowed with Riemannian quotient geometries”. In: *Computational Statistics* 29.3-4 (2014), pp. 569–590.

[118] G. Hurlbert. “A short proof of the Birkhoff-von Neumann Theorem”. In: *preprint (unpublished)* (2008).

[119] C. J. Lin. “Projected gradient methods for nonnegative matrix factorization”. In: *Neural computation* 19.10 (2007), pp. 2756–2779.

[120] N. Gillis and F. Glineur. “Accelerated multiplicative updates and hierarchical ALS algorithms for nonnegative matrix factorization”. In: *Neural computation* 24.4 (2012), pp. 1085–1105.

[121] MIT Center For Biological and Computation Learning. “CBCL Face Database 1”. In: *http://www.ai.mit.edu/projects/cbcl* 1.1 (2015).

[122] F. S. Samaria and A. C. Harter. “Parameterisation of a stochastic model for human face identification”. In: *Proc. of the 2nd IEEE Workshop on Applications of Computer Vision (ACV’ 1994), Sarasota, FL, USA*. 1994, pp. 138–142.

[123] D. D. Lee and H. S. Seung. “Learning the parts of objects by non-negative matrix factorization”. In: *Nature* 401.6755 (1999), p. 788.

[124] E. J. Candes, T. Strohmer, and V. Voroninski. “Phaselift : Exact and stable signal recovery from magnitude measurements via convex programming”. In: *Communications on Pure and Applied Mathematics* 66.8 (2013), pp. 1241–1274.

[125] P. Netrapalli, P. Jain, and S. Sanghavi. “Phase retrieval using alternating minimization”. In: *Proc. of the Advances in Neural Information Processing Systems (NIPS’ 2013), NV, USA*. 2013, pp. 2796–2804.

[126] E. J. Candes, X. Li, and M. Soltanolkotabi. “Phase retrieval via Wirtinger flow: Theory and algorithms”. In: *IEEE Transactions on Information Theory* 61.4 (2015), pp. 1985–2007.

[127] G. Wang, G. B. Giannakis, and Y. C. Eldar. “Solving systems of random quadratic equations via truncated amplitude flow”. In: *IEEE Transactions on Information Theory* 64.2 (2017), pp. 773–794.

[128] M. Dierolf et al. “Ptychographic X-ray computed tomography at the nanoscale”. In: *Nature* 467.7314 (2010), p. 436.

[129] R. P. Millane. “Phase retrieval in crystallography and optics”. In: *JOSA A* 7.3 (1990), pp. 394–411.

[130] A. Douik et al. “Manifold Optimization for High Accuracy Spacial Location Estimation Using Ultrasound Waves”. In: *Submitted to IEEE Transactions on Signal Processing* (Dec. 2020).

[131] R. M. Narayanan and M. Dawood. “Doppler estimation using a coherent ultrawide-band random noise radar”. In: *IEEE Transactions on Antennas and Propagation* 48.6 (2000), pp. 868–878.

[132] M. Cypriani et al. “Open wireless positioning system: A Wi-Fi-based indoor positioning system”. In: *Proc. of the 70th IEEE Vehicular Technology Conference Fall (VTC-F’09), Anchorage, AK, USA*. IEEE. 2009, pp. 1–5.

[133] R. H. Rasshofer and K. Gresser. “Automotive Radar and Lidar Systems for Next Generation Driver Assistance Functions.” In: *Advances in Radio Science* 3 (2005).

[134] M. Kushwaha et al. “Sensor node localization using mobile acoustic beacons”. In: *Proc. of the IEEE International Conference on Mobile Adhoc and Sensor Systems Conference (MASS’05), Washington, DC, USA*. IEEE. 2005, pp. 9–15.

[135] F. Seco et al. “A survey of mathematical methods for indoor localization”. In: *Proc. of the IEEE International Symposium on Intelligent Signal Processing (WISP’09), Budapest, Hungary*. IEEE. 2009, pp. 9–14.

[136] F. Zafari, A. Gkelias, and K. K. Leung. “A survey of indoor localization systems and technologies”. In: *IEEE Communications Surveys & Tutorials* 21.3 (2019), pp. 2568–2599.

[137] Z. Yang, Z. Zhou, and Y. Liu. “From RSSI to CSI: Indoor Localization via Channel Response”. In: *ACM Computing Surveys (CSUR)* 46.2 (2013), pp. 1–32.

[138] S. Kumar et al. “Accurate Indoor Localization With Zero Start-up Cost”. In: *Proc. of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom’14), Maui, Hawaii, USA*. 2014, pp. 483–494.

[139] S. Wright and J. Nocedal. “Numerical optimization”. In: *Springer Science* 35.67-68 (1999), p. 7.

[140] A. Ward, A. Jones, and A. Hopper. “A new location technique for the active office”. In: *IEEE Personal communications* 4.5 (1997), pp. 42–47.

[141] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. “The cricket location-support system”. In: *Proc. of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom’00), Boston, Massachusetts, USA*. 2000, pp. 32–43.

[142] Y. Fukuju et al. “DOLPHIN: An autonomous indoor positioning system in ubiquitous computing environment”. In: *Proc. of the IEEE Workshop on Software Technologies for Future Embedded Systems (WSTFES’03), Hokkaido, Japana*. IEEE. 2003, pp. 53–56.

[143] M. M. Saad et al. “High-accuracy reference-free ultrasonic location estimation”. In: *IEEE Transactions on Instrumentation and Measurement* 61.6 (2012), pp. 1561–1570.

[144] J. R. Gonzalez and C. J. Bleakley. “High-precision robust broadband ultrasonic location and orientation estimation”. In: *IEEE Journal of selected topics in Signal Processing* 3.5 (2009), pp. 832–844.

[145] W. H. Foy. “Position-location solutions by Taylor-series estimation”. In: *IEEE Transactions on Aerospace and Electronic Systems* 2 (1976), pp. 187–194.

[146] M. Hazas and A. Ward. “A novel broadband ultrasonic location system”. In: *Proc. of the International Conference on Ubiquitous Computing (UbiComp’02), Goteborg, Sweden*. Springer. 2002, pp. 264–280.

[147] M. Hazas and A. Hopper. “Broadband ultrasonic location systems for improved indoor positioning”. In: *IEEE Transactions on mobile Computing* 5.5 (2006), pp. 536–547.

[148] R. H. Byrd, J. C. Gilbert, and J. Nocedal. “A trust region method based on interior point techniques for nonlinear programming”. In: *Mathematical programming* 89.1 (2000), pp. 149–185.

[149] R. H. Byrd, M. E. Hribar, and J. Nocedal. “An interior point algorithm for large-scale nonlinear programming”. In: *SIAM Journal on Optimization* 9.4 (1999), pp. 877–900.

[150] R. A. Waltz et al. “An interior algorithm for nonlinear optimization that combines line search and trust region steps”. In: *Mathematical programming* 107.3 (2006), pp. 391–408.

[151] J. Nocedal and S. Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[152] R. H. Byrd, R. B. Schnabel, and G. A. Shultz. “A trust region algorithm for nonlinearly constrained optimization”. In: *SIAM Journal on Numerical Analysis* 24.5 (1987), pp. 1152–1170.

[153] N. Agarwal et al. “Adaptive regularization with cubics on manifolds”. In: *arXiv preprint arXiv:1806.00065* (2018).

[154] M. H. AlSharif et al. “Zadoff-Chu coded ultrasonic signal for accurate range estimation”. In: *Proc. of the 25th European Signal Processing Conference (EUSIPCO’17), Kos Island, Greece*. IEEE. 2017, pp. 1250–1254.

[155] P. J. G. Teunissen. “The affine constrained GNSS attitude model and its multivariate integer least-squares solution”. In: *Journal of Geodesy* 86.7 (July 2012), pp. 547–563.

[156] P. J. G. Teunissen. “Integer least-squares theory for the GNSS compass”. In: *Journal of Geodesy* 84.7 (July 2010), pp. 433–447.

[157] J. L. Crassidis, F. L. Markley, and E. G. Lightsey. “Global Positioning System integer ambiguity resolution without attitude knowledge”. In: *Journal of Guidance, Control and Dynamics* 22.2 (Mar. 1999), pp. 212–218.

[158] C. Chun and F. C. Park. “Dynamics-based attitude determination using the Global Positioning System”. In: *Journal of Guidance, Control and Dynamics* 24.3 (May 2001), pp. 466–473.

[159] B. Wang et al. “A motion-based integer ambiguity resolution method for attitude determination using the Global Positioning System (GPS)”. In: *Measurement Science and Technology* 21.6 (Apr. 2010), p. 065102.

[160] P. J. G. Teunissen. “The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation”. In: *Journal of Geodesy* 70.1 (Nov. 1995), pp. 65–82.

[161] C. Park et al. “Efficient ambiguity resolution using constraint equation”. In: *Proc. of the IEEE Position Location and Navigation Symposium (PLANS’ 1996), Atlanta, GA, USA*. Atlanta, Georgia, USA, Apr. 1996, pp. 277–284.

[162] J. C. Juang and G. S. Huang. “Development of GPS-based attitude determination algorithms”. In: *IEEE Transactions on Aerospace and Electronic Systems* 33.3 (July 1997), pp. 968–976.

[163] S. Purivigraipong et al. “Resolving Integer Ambiguity of GPS Carrier Phase Difference”. In: *IEEE Transactions on Aerospace and Electronic Systems* 46.2 (May 2010), pp. 832–847.

[164] X. W. Chang, X. Yang, and T. Zhou. “MLAMBDA: a modified LAMBDA method for integer least-squares estimation”. In: *GPS Solutions* 79.9 (Dec. 2005), pp. 552–565.

[165] B. Wang et al. “A constrained LAMBDA method for GPS attitude determination”. In: *GPS Solutions* 13.2 (Mar. 2009), pp. 97–107.

[166] C. Gunther and P. Henkel. “Integer ambiguity estimation for satellite navigation”. In: *IEEE Transactions on Signal Processing* 60.7 (July 2012), pp. 3387–3393.

[167] G. Blewitt. “Basics of the GPS technique: observation equations”. In: *Geodetic applications of GPS* (1997), pp. 10–54.

[168] T. Ballal and C. J. Bleakley. “GNSS instantaneous ambiguity resolution and attitude determination exploiting the receiver antenna configuration”. In: *IEEE Transactions on Aerospace and Electronic Systems* 50.3 (July 2014), pp. 2061–2069.

[169] W. Chen and X. Li. “Success rate improvement of single epoch integer least-squares estimator for the GNSS attitude/short baseline applications with common clock scheme”. In: *Acta Geodaetica et Geophysica* 49.3 (Sept. 2014), pp. 295–312.

[170] W. Chen et al. “Formal Uncertainty and Dispersion of Single and Double Difference Models for GNSS-Based Attitude Determination”. In: *Sensors* 17.2 (Feb. 2017), p. 408.

[171] L. Wanninger. “Carrier-phase inter-frequency biases of GLONASS receivers”. In: *Journal of Geodesy* 86.2 (Feb. 2012), pp. 139–148.

[172] G. Giorgi and P. J. G. Teunissen. “Carrier phase GNSS attitude determination with the Multivariate Constrained LAMBDA method”. In: *Proc. of the IEEE Aerospace Conference (AeroConf’ 2010), Big Sky, MT, USA*. Vol. 1. 1. Mar. 2010, pp. 1–12. DOI: `10.1109/AERO.2010.5446910`.

[173] T. Ballal and C. J. Bleakley. “Phase-Difference Ambiguity Resolution for a Single-Frequency Signal”. In: *IEEE Signal Processing Letters* 15.1 (Dec. 2008), pp. 853–856.

[174] T. Ballal and C. J. Bleakley. “Phase-difference ambiguity resolution for a single-frequency signal in the near-field using a receiver triplet”. In: *IEEE Transactions on Signal Processing* 58.11 (Aug. 2010), pp. 5920–5926.

[175] S. Alban. “Design and performance of a robust GPS/INS attitude system for automobile applications”. PhD thesis. Standford, CA, USA: Stanford University, 2004.

[176] P. J. G. Teunissen. “A-PPP: Array-Aided Precise Point Positioning With Global Navigation Satellite Systems”. In: *IEEE Transactions on Signal Processing* 60.6 (Mar. 2012), pp. 2870–2881.

[177] Y. Yang. “Attitude determination using Newton’s method on Riemannian manifold”. In: *Proc. of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 229.14 (2015), pp. 2737–2742. DOI: 10.1177/0954410015591196.

[178] J. Wu et al. “Fast Linear Quaternion Attitude Estimator Using Vector Observations”. In: *IEEE Transactions on Automation Science and Engineering* 15.1 (Jan. 2018), pp. 307–319.

[179] G. Giorgi and P. J. G. Teunissen. “Carrier phase GNSS attitude determination with the multivariate constrained LAMBDA method”. In: *Proc. of the IEEE Aerospace Conference (AeroConf’ 2010), Big Sky, MT, USA*. IEEE. 2010, pp. 1–12.

[180] P. J. Buist et al. “Multiplatform instantaneous GNSS ambiguity resolution for triple-and quadruple-antenna configurations with constraints”. In: *International Journal of Navigation and Observation* 2009 (2009).

[181] S. Verhagen, B. Li, and M. Geodesy. *LAMBDA software package: Matlab implementation, Version 3.0*. Delft University of Technology and Curtin University, Perth, Australia, 2012.

[182] P. Cederholm. “Statistical characteristics of L1 carrier phase observations from four low-cost GPS receivers”. In: *Nordic Journal of Surveying and Real Estate Research* 7.1 (Dec. 2010), pp. 58–75.

QUESTIONNAIRE

CONSENT FORM