Riemannian Optimization Algorithms for Applications and Their Theoretical Properties

March 2024

LAI ZHIJIAN

Riemannian Optimization Algorithms for Applications and Their Theoretical Properties

Graduate School of Science and Technology

Degree Programs in Systems and Information Engineering

University of Tsukuba

March 2024

LAI ZHIJIAN

I would like to dedicate this thesis to my loving parents.

Acknowledgements

First and foremost, I wish to extend my deepest gratitude to my family, with a special mention to my parents and brother. Their unwavering support for my academic aspirations, despite the significant time I spent away from home, has been my driving force. Your understanding and unwavering support have been an endless source of motivation.

I want to express my heartfelt appreciation to Prof. Akiko Yoshise for her unwavering guidance and mentorship throughout my five-year journey at the University of Tsukuba. Her support and direction have played a pivotal role in shaping my academic path. My sincere thanks go to Prof. Ying Miao for his invaluable insights and advice regarding my future career. Furthermore, I would like to thank Prof. Hiroyuki Sato of Kyoto University, who provided expert insights into the Riemannian optimization. I would also like to thank Prof. Takahito Kuno and Prof. Masahiro Hachimori for their careful feedback and constructive suggestions during the preparation of my thesis and defense.

To all my friends, colleagues, and mentors who have contributed to my academic growth and supported me throughout this journey, I offer my sincere appreciation. This thesis is dedicated to all those mentioned above and to everyone who has played a vital role in my quest for knowledge and personal development.

LAI ZHIJIAN
March, 2024

Abstract

Riemannian Optimization (RO) is a vibrant and important research area in the field of optimization theory, which focuses on optimizing real-valued functions over Riemannian manifolds. The manifolds are characterized as smooth curved spaces that generalize Euclidean spaces. The aim of this dissertation is to discuss recent advances and applications of Riemannian optimization.

Despite the relatively short history of RO, both its theoretical research and practical applications have experienced rapid growth over the past 20 years. The main trend in RO's theoretical research is to extend various classical and well-established optimization algorithms (e.g., gradient descent, Newton method, etc.) from Euclidean spaces to Riemannian manifolds. In contrast to the familiar Euclidean spaces, Riemannian manifolds are more generalized spaces equipped with all the necessary geometric tools for the optimization algorithms. In practice, Riemannian manifolds are often represented as the feasible region of an optimization problem, while being a subset of a Euclidean space. By utilizing the Riemannian geometry of that feasible region itself, Riemannian algorithms tend to be more efficient than Euclidean algorithms.

Existing researches of RO have focused on optimization problems with a smooth objective function over a single manifold. However, as the range of applications increases, this type of optimization problem sometimes does not satisfy diverse demands well, which leads to algorithm development efforts for the variants of Riemannian optimization. This dissertation investigates two variants of RO, each of which addresses different challenges in practical applications. The first variant, Nonsmooth Riemannian Optimization (NRO), is concerned with optimization problems characterized by a nonsmooth objective function. The second variant, Constrained Riemannian Optimization (CRO), addresses optimization problems on non-single manifold, i.e., problems involve additional constraints that cannot form a manifold. Both NRO and CRO have proven to be more applicable and closely related to real-world applications, thus extending the scope and effectiveness of RO.

For NRO problems, we will propose a generalized framework of Riemannian smoothing method to solve these problems, ensuring efficient convergence to the limiting stationary point. Our framework is not only user-friendly but also synergizes seamlessly with existing solvers, like "Manopt", thereby facilitating fast and straightforward code implementation. Numerical experiments demonstrate the efficiency of the method. In particular, we apply our method to the CP factorization problem, which is an open problem in the theory of conic optimization. Numerical experiments confirm that our method is particularly superior for large-scale CP factorization problems.

For CRO problem, we will propose the Riemannian interior point method, which is an extension of the classical primal-dual interior point method to Riemannian manifold. We prove that our method holds locally superlinear and quadratically convergence, and that global convergence can be achieved by combining it with classical linear search. Numerical experiments confirm the stability and efficiency of

our proposal. To our knowledge, this is the first study to apply the primal-dual interior point method to the constrained optimization problem on Riemannian manifolds. One significant contribution is that we establish many essential foundational concepts for the general interior point method in the Riemannian context, such as the KKT vector field and its covariant derivative. In addition, we build the first algorithm framework for the Riemannian version of the interior point method. These contributions will be useful in the future, especially in the development of more advanced interior point methods.

In conclusion, Riemannian optimization is a vigorous and rapidly growing field that has seen significant advances in both theory and practical applications in recent years. The exploration of variants such as NRO and CRO has expanded the potential and applicability of this optimization framework, opening up exciting opportunities for future development and real-world problem solutions.

LAI ZHIJIAN
March, 2024

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

Set

$\mathbb{N}$	Set of natural numbers
$\mathbb{R}$	Set of real numbers
$\mathbb{R}_+$	Set of nonnegative real numbers
$\mathbb{R}_{++}$	Set of positive real numbers
$\mathbb{R}^n, \mathbb{R}^m, \mathbb{R}^l$	Set of real column vectors of size n, m, l
$\mathbb{R}^n_+$	Set of nonnegative real column vectors of size n, namely, nonnegative orthant
$\mathbb{R}^{m\times n}$	Set of $m \times n$ real matrices
$\mathbb{R}^{m\times n}_+$	Set of $m \times n$ nonnegative matrices
$\mathbb{C}$	Set of complex numbers
$\mathbb{C}^{m\times n}$	Set of $m \times n$ complex matrices
$\mathrm{S}(n)$	Set of $n \times n$ real symmetric matrices
$\mathrm{S}^+(n)$	Cone of $n \times n$ real symmetric positive semidefinite matrices
$\mathrm{CP}(n)$	Cone of $n \times n$ completely positive matrices
$\mathrm{COP}(n)$	Cone of $n \times n$ copositive matrices
$\mathrm{N}(n)$	Cone of $n \times n$ symmetric nonnegative matrices
$\mathrm{Skew}(n)$	Set of $n \times n$ real skew-symmetric matrices
$\mathrm{Proj}_S(x)$	Project x onto the set S
$\mathrm{conv}(S)$	Convex hull of the set S
$\mathrm{int}(S)$	Interior of the set S
$\mathrm{bd}(S)$	Boundary of the set S

Vector

$z_i, (z_k)_i$	i-th entry of column vectors z and z_k
$\|z\|_0$	ℓ_0 norm of vector z, the number of nonzero entries of z, despite the fact that it is not a well-defined norm
$\|z\|_1$	ℓ_1 norm of vector z, $\|z\|_1 := \sum_i \|z_i\|$
$\|z\|_2$	ℓ_2 norm of vector z, $\|z\|_2 := \sqrt{\sum_i \|z_i\|^2}$
$\mathrm{Diag}(z)$	Diagonal matrix whose diagonal elements are given by the vector z

Matrix

$X_{ij}, (X_k)_{ij}$	Entry of the matrices X and X_k at the i-th row and j-th column
$X_{:,j}$	j-th column of the matrix X

X^T	Transpose of the matrix X
X^*	Conjugate transpose of the matrix X
$X \succeq 0$	X is a real symmetric positive semidefinite matrix
$X \succ 0$	X is a real symmetric positive definite matrix
$X \geq 0$	X has all entries nonnegative
$X > 0$	X has all entries positive
$\|X\|_{\mathrm{F}}$	Frobenius norm of the matrix X, $\|X\|_{\mathrm{F}} := \sqrt{\sum_{i,j} \|X_{ij}\|^2}$
$\|X\|_1$	ℓ_1 norm applied to the matrix X, $\|X\|_1 := \sum_{i,j} \|X_{ij}\|$
$\|X\|_2$	Spectral norm of the matrix X, namely, the largest singular value of X
$\mathrm{rank}(X)$	Rank of the matrix X
$\mathrm{trace}(X)$	Trace of the square matrix X (sum of the diagonal entries)
$\det(X)$	Determinant of the square matrix X
$\mathrm{diag}(X)$	Column vector extracted from the daigonal of the square matrix X
$\mathrm{span}(X)$	Subspace spanned by the columns of the matrix X
$\mathrm{cp}(X)$	cp-rank of the real symmetric matrix X
$\mathrm{cp}^+(X)$	cp-plus-rank of the real symmetric matrix X
cp_n	A special number defined in Theorem 13

Constant

e	Euler's number
I, I_n	Identity matrix of size n (or of size indicated by context)
$\mathbf{1}, \mathbf{1}_n$	All-ones column vector of size n (or of size indicated by context)
$\mathbf{E}$	All-ones matrix with proper size by context
$\{\mathbf{e}_i\}_{i=1}^n$	Standard basis of the n-dimensional real space $\mathbb{R}^n$, namely, the columns of I_n
$0, 0_m, 0_{m\times n}$	Number 0 or the zero vector or zero matrix (or of size indicated by context)
$\hat{e}$	A special vector field as defined in (5.26)

Linear Algebra

$\mathcal{E}, \mathcal{E}'$	Finite-dimensional vector spaces
$\langle \cdot, \cdot \rangle_{\mathcal{E}}$	An inner product defined on $\mathcal{E}$
$\|v\|_{\mathcal{E}}$	Induced norm on $\mathcal{E}$, $\|v\|_{\mathcal{E}} := \sqrt{\langle v, v \rangle_{\mathcal{E}}}$ for $v \in \mathcal{E}$
$\hat{v}$	Vector representation of some element $v \in \mathcal{E}$ w.r.t. some basic of $\mathcal{E}$
$\mathcal{A}, \mathcal{B}, \mathcal{P}$	General linear operators between vector spaces
(In the following, $\mathcal{A}\colon \mathcal{E} \to \mathcal{E}'$ is a linear operator)	
$\mathcal{A}(v), \mathcal{A}v$	Call of $\mathcal{A}$ on vector $v \in \mathcal{E}$
$\mathcal{A}^*$	Adjoint operator of $\mathcal{A}$, note that $\mathcal{A}^*\colon \mathcal{E}' \to \mathcal{E}$ is also linear
$\|\mathcal{A}\|$	Operator norm of $\mathcal{A}$, $\|\mathcal{A}\| := \sup\{\|\mathcal{A}v\|_{\mathcal{E}'} \mid v \in \mathcal{E}, \|v\|_{\mathcal{E}} = 1\} = \sup\{\|\mathcal{A}v\|_{\mathcal{E}'} \mid v \in \mathcal{E}, \|v\|_{\mathcal{E}} \leq 1\}$
$\hat{\mathcal{A}}$	Matrix representation of $\mathcal{A}$ w.r.t. some bases of $\mathcal{E}$ and $\mathcal{E}'$
$\ker \mathcal{A}$	Kernel space or null space of $\mathcal{A}$, $\ker \mathcal{A} := \{v \in \mathcal{E} : \mathcal{A}v = 0\}$

Calculus

$f \circ g, fg$	Function composition, $(fg)(x) = (f \circ g)(x) = f(g(x))$

$\mathbf{H}_f(x)$	Hessian matrix $\mathbf{H}_f(x) \in \mathbb{R}^{n\times n}$ of the function $f\colon \mathbb{R}^n \to \mathbb{R}$ at $x \in \mathbb{R}^n$
$\mathbf{J}_F(x)$	Jacobian matrix $\mathbf{J}_F(x) \in \mathbb{R}^{m\times n}$ of the function $F\colon \mathbb{R}^n \to \mathbb{R}^m$ at $x \in \mathbb{R}^n$
$\max(\cdot), \min(\cdot)$	Functions to obtain the maximum or minimum value of the arguments
$\exp(\cdot)$	Classical exponential function, or matrix exponential of square matrix
$\log(\cdot)$	Classical logarithm function
C^1	Continuously differentiable function
C^2	Twice continuously differentiable function
C^s	s times continuously differentiable function
C^∞	Infinitely differentiable, namely, smooth function
$B_r(x)$	Open ball with center $x \in \mathbb{R}^n$ and radius $r > 0$, $B_r(x) := \{y \in \mathbb{R}^n : \|x-y\|_2 < r\}$
$C^1(B_r(x))$	Set of all C^1 real-valued functions defined on $B_r(x)$
(In the following, $f\colon \mathbb{R}^n \to \mathbb{R}$ is lower semi-continuous)	
$\tilde{f}$	Smoothing function of f, see Definition 36
lse	Log-sum-exp function, a smoothing function of $\max(\cdot)$, see Example 20
$\mu_k \downarrow 0$	Positive sequence $\{\mu_k\}$ converging to 0 from the right
Δ^{n-1}	Unit simplex, $\Delta^{n-1} := \{x \in \mathbb{R}^n : \sum_{i=1}^n x_i = 1, x_i \geq 0\}$
$\hat{\partial} f(x)$	Regular (or Fréchet) subdifferential of f at x
$\partial f(x)$	Limiting subdifferential of f at x
$\partial^\circ f(x)$	Clarke subdifferential of f at x
$\hat{\partial}_{\mathrm{R}} f(x)$	Riemannian regular (or Fréchet) subdifferential of f at x
$\partial_{\mathrm{R}} f(x)$	Riemannian limiting subdifferential of f at x
$G_{\tilde{f}}(x)$	Subdifferential of f associated with $\tilde{f}$ at x
$G_{\tilde{f},\mathrm{R}}(x)$	Riemannian subdifferential of f associated with $\tilde{f}$ at x

Riemannian Geometry

$\mathcal{M}, \mathcal{N}$	Smooth, finite-dimensional (usually Riemannian) manifolds
$\mathcal{N}$	Only in Chapter 5-7, it denotes special product manifold, $\mathcal{N} := \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$, see (2.38)
Sp^{n-1}	Sphere manifold, $\mathrm{Sp}^{n-1} := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$
$\mathrm{St}(n,k)$	Stiefel manifold, $\mathrm{St}(n,k) := \{X \in \mathbb{R}^{n\times k} : X^T X = I_k\}$
$\mathrm{O}(r)$	Orthogonal group, $\mathrm{O}(r) := \{X \in \mathbb{R}^{r\times r} : X^T X = I_n\}$
$\mathrm{Ob}(n,k)$	Oblique manifold, $\mathrm{Ob}(n,k) := \{X \in \mathbb{R}^{n\times k} :$ each column of X has unit l_2 norm$\}$
$\mathrm{Fr}(m,n,r)$	Fixed-rank manifold, $\mathrm{Fr}(m,n,r) := \{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) = k\}$
$\dim \mathcal{M}$	Dimension of the manifold $\mathcal{M}$
$\mathcal{U}, \mathcal{V}$	Some open subsets, neighborhood on manifold, or domain of a chart
$\mathfrak{F}(\mathcal{M})$	Set of all smooth real-valued functions defined on $\mathcal{M}$
$\mathfrak{F}_x(\mathcal{M})$	Set of all smooth real-valued functions defined on a neighborhood around $p \in \mathcal{M}$
$T_x\mathcal{M}$	Tangent space at point $p \in \mathcal{M}$
ξ, η, ζ	Tangent vectors
0_x	Zero element of $T_x\mathcal{M}$
$\langle \cdot, \cdot \rangle_x$	Inner product defined on $T_x\mathcal{M}$
$\|\xi\|_x$	Induced norm on $T_x\mathcal{M}$

$\{E_i\}_{i=1}^n$	Orthonormal basis of $T_x\mathcal{M}$, or orthonormal local frame around x
$\mathrm{Proj}_x(\cdot)$	Orthogonal projection onto $T_x\mathcal{M}$
Id, $\mathrm{Id}_{T_x\mathcal{M}}$	Identity map defined on $T_x\mathcal{M}$ (or on the domain indicated by context)
D	Differential operator
$T\mathcal{M}$	Tangent bundle of $\mathcal{M}$, $T\mathcal{M} := \{(x, v) : x \in \mathcal{M}, v \in T_x\mathcal{M}\}$
$T\mathcal{M} \oplus T\mathcal{M}$	Whitney sum of tangent bundles, $T\mathcal{M} \oplus T\mathcal{M} := \{(x, \eta, \xi) : x \in \mathcal{M}, \eta, \xi \in T_x\mathcal{M}\}$
F, U, V, W	Vector fields
$\mathfrak{X}(\mathcal{M})$	Set of all smooth vector fields on $\mathcal{M}$
R	Retractions on a manifold
$\bar{\mathrm{R}}$	Retractions on the special product manifold $\mathcal{N}$
$\mathrm{egrad}\, f(x)$	Classical (Euclidean) gradient
$\mathrm{grad}\, f(x)$	Riemannian gradient
∇	Connection on a manifold
$[U, V]$	Lie bracket of vector fields U and V
$\nabla V(x)$	Covariant derivative of the vector field V at point x
$\mathrm{eHess}\, f(x)$	Classical (Euclidean) Hessian
$\mathrm{Hess}\, f(x)$	Riemannian Hessian
I	Some real interval, $I \in \mathbb{R}$
$\mathfrak{F}(I)$	Set of all smooth real-valued functions defined on I
$\gamma(t), c(t)$	Curves on a manifold, e.g., $\gamma\colon \mathcal{M}$
$\mathfrak{X}(c)$	Set of all smooth vector fields along the curve c
$\dot{c}(t)$	Velocity of a curve at time t
$\ddot{c}(t)$	Acceleration of a curve at time t
$\frac{\mathrm{D}}{\mathrm{d}t}$	Induced covariant derivative (induced by ∇)
Exp	Exponential map on a manifold
$\bar{\mathrm{Exp}}$	Exponential map on the special product manifold $\mathcal{N}$
$d(x, y)$	Distance between points x and y on a manifold
P_γ, $\mathrm{P}_\gamma^{t_1 \to t_0}$	Parallel transport along a curve γ on a manifold
T	Vector transport on a manifold
$\bar{\mathrm{T}}$	Vector transport on the special product manifold $\mathcal{N}$

Miscellaneous

$u_k = O(v_k)$	Big-O notation, there exists $M > 0$ such that $u_k \leq M v_k$ for all sufficiently large k
$u_k = o(v_k)$	Little-o notation, the sequence of ratios $\{u_k/v_k\}$ approaches zero
$x \sim y$	Equivalence relation evaluated for two objects x and y
$[x]$	Equivalence class of x for the equivalence relation $\sim$
$\mathbb{E}$	Equality index set, $\mathbb{E} := \{1, 2, \ldots, l\}$
$\mathbb{I}$	Inequality index set, $\mathbb{I} := \{1, 2, \ldots, m\}$
$\mathbb{A}(x)$	Active set at point x, $\mathbb{A}(x) := \{i : g_i(x) = 0\}$, see (CRO)
w	$w = (x, y, z, s) \in \mathcal{N}$, see (2.38)
$\mathcal{L}(x, y, z)$	Lagrangian of (CRO), $\mathcal{L}(x, y, z) := f(x) + \sum_{j=1}^l y_j h_j(x) + \sum_{i=1}^m z_i g_i(x)$
$\mathrm{grad}_x\, \mathcal{L}(x, y, z)$	Riemannian gradient of Lagrangian with respect to the manifold variable x

$\mathrm{Hess}_x\, \mathcal{L}(x, y, z)$	Riemannian Hessian of Lagrangian with respect to the manifold variable x
$\mathcal{C}(x, y, z)$	critical cone, see (5.10)
$\mathcal{H}_x$	A special linear operator defined in (5.19), which depends on x
$\mathcal{G}_x$	A special linear operator defined in (5.19), which depends on x
Ψ	A special linear operator defined in (5.34), which depends on w
$\mathcal{A}_w$	A special linear operator defined in (5.33), which depends on w
$\mathcal{T}$	A special linear operator defined in (5.32), which depends on w

Acronyms and abbreviations

ACQ	Abadie's Constraint Qualification
ADMM	Alternating Directions Method of Multipliers method
AP	Additional Properties
BFGS	Broyden-Fletcher-Goldfarb-Shanno algorithm
CEO	Constrained Euclidean Optimization
CG	Conjugate Gradient method
CP	Completely Positive matrix
CQs	Constraint Qualifications
CR	Conjugate Residual method
CRO	Constrained Riemannian Optimization
EIPM	Euclidean Interior Point Method
EO	Euclidean Optimization
FSV	Finding the Sparsest Vector
GCQ	Guignard's Constraint Qualification
IPM	Interior Point Methods
KKT	Karush-Kuhn-Tucker conditions
LICQ	Linear Independence Constraint Qualification
MADS	Mesh-Adapted Direct Search algorithm
ManPG	Manifold Proximal Gradient method
MFCQ	Mangasarian-Fromovitz Constraint Qualification
NLRM	Nonnegative Low-Rank Matrix approximation
NMF	Nonnegative Matrix Factorization
NRO	Nonsmooth Riemannian Optimization
RALM	Riemannian Augmented Lagrangian Method
REPM	Riemannian Exact Penalty Method
RIPM	Riemannian Interior Point Method
RMSE	Root Mean Square Error
RO	Riemannian Optimization
RSM	Riemannian Smoothing Method
RSQP	Riemannian Sequential Quadratic Programming method
SD	Steepest Descent method
SOCP	Second-Order Cone Problem
SONC	Second-Order Necessary Conditions

SOSC	Second-Order Sufficient Conditions
SQP	Sequential Quadratic Programming
TR	Trust Regions method
UEO	Unconstrained Euclidean Optimization
URO	Unconstrained Riemannian Optimization

Part I

Preliminary

Chapter 1

Overview of Several Types of Riemannian Optimizations

Part	Section
Part 1. Riemannian Optimizations (RO)	1.1 Optimization from Euclidean Spaces to Riemannian Manifolds
Part 2. Two variants of RO	1.2 Nonsmooth Riemannian Optimization 1.3 Constrained Riemannian Optimization
Part 3. Outline	1.4 Summary and Thesis Structure

1.1 Optimization from Euclidean Spaces to Riemannian Manifolds

1.1.1 Euclidean Optimization

Mathematical optimization is an important field in applied mathematics that involves selecting the best element from a set of available alternatives. The aim of this process is to efficiently find the solution that minimizes or maximizes a given objective function (also known as a cost function), subject to a set of constraints. Mathematical optimizations are often formally represented as follows: given a function $f\colon \mathbb{R}^n \to \mathbb{R}$, the general form of a (Euclidean) optimization problem is

$$\begin{aligned} &\min\ f(x) \\ &\text{s.t. } x \in S. \end{aligned} \tag{EO}$$

Here, $x = [x_1, x_2, \ldots, x_n]^T$ is an n-dimensional vector in the Euclidean space $\mathbb{R}^n$, and the feasible region $S \subset \mathbb{R}^n$ is the feasible set of all possible solutions. Typically, we default to thinking of feasible region S as a subset of Euclidean space $\mathbb{R}^n$. This covers almost all the problem patterns. Matrix spaces or general linear spaces are also equivalent to the above formulation. Thus, the traditional and classical optimization theory actually consider an optimization problem underlying the Euclidean space. To distinguish it from the Riemannian optimization of the later context, we specifically call it Euclidean Optimization (EO).

Problem 1 (Unconstrained Euclidean Optimization (UEO))**.** In unconstrained optimization, the set S is the entire space $\mathbb{R}^n$. The problem can be written as:

$$\begin{aligned} &\min\ f(x) \\ &\text{s.t. } x \in \mathbb{R}^n. \end{aligned} \tag{UEO}$$

The Unconstrained Euclidean Optimization (UEO) do not have constraints on variable, providing more freedom in finding optimal solutions. The line search method, as shown in Algorithm 1, is a common framework to solve this problem. The objective function $f\colon \mathbb{R}^n \to \mathbb{R}$ is assumed to be smooth. The line search method seeks to find a sequence $\{x_k\}$ that converges to a minimizer of the function f.

Algorithm 1: General Line Search Method for (UEO)

Input: An objective function f defined on $\mathbb{R}^n$, an initial point $x_0 \in \mathbb{R}^n$.
Output: Sequence $\{x_k\} \subset \mathbb{R}^n$.
Set $k \to 0$;
while *stopping criterion not satisfied* **do**
1. Compute a search direction $d_k \in \mathbb{R}^n$;
2. Compute a step size $t_k > 0$;
3. Compute the next point as $x_{k+1} := x_k + t_k d_k$;
4. $k \to k + 1$;

end

How the descent direction d_k is chosen determines the different optimization algorithms:
- steepest gradient descent method: $d_k = -\nabla f(x_k)$ where $\nabla f(x_k)$ is the gradient of f at x_k.
- Newton method: $d_k = -\left[\nabla^2 f(x_k)\right]^{-1} \nabla f(x_k)$ where $\nabla^2 f(x_k)$ is the Hessian of f at x_k.
- quasi-Newton method: $d_k = -B_k^{-1} \nabla f(x_k)$ where B_k is an approximation to the Hessian.

The success of linear search frameworks depends to a large extent on the selection strategy for the direction and step size. Depending on these choices and the characteristics of the objective function, the convergence properties of the linear search framework are different.

Problem 2 (Constrained Euclidean Optimization (CEO))**.** In constrained optimization, the set S is a proper subset of $\mathbb{R}^n$ defined by equality and inequality constraints. The problem is usually expressed as:

$$\begin{aligned} \min\quad & f(x) \\ \text{s.t.}\quad & g_i(x) = 0,\ i = 1, 2, \ldots, m, \\ & h_j(x) \le 0,\ j = 1, 2, \ldots, l, \\ & x \in \mathbb{R}^n, \end{aligned} \tag{CEO}$$

where $g_i\colon \mathbb{R}^n \to \mathbb{R}$ are equality constraints and $h_j\colon \mathbb{R}^n \to \mathbb{R}$ are inequality constraints.

Compared to the unconstrained case, the Constrained Euclidean Optimization (CEO) introduces specific constraints or bounds on the variables x, which leads to a more challenging optimization problem. Such problems include linear programming, quadratic programming, convex optimization, and general nonlinear nonconvex optimization. The solution x must satisfy all constraints to be considered valid, or feasible. The line search framework of Algorithm 1 is unable to solve this problem since the new iterate

$x_{k+1} := x_k + t_k d_k$ may not satisfy the constraints (even if x_k satisfies). Algorithms for constrained optimization often involve techniques that handle constraints explicitly, such as augmented Lagrangian methods, barrier methods, penalty methods, Sequential Quadratic Programming (SQP) method and Interior Point Methods (IPM). Each method provides a ingenious way of balancing the objective function minimization with the need to adhere to the constraints, ultimately guiding the algorithm towards feasible and optimal solutions.

Next, let we consider a new thought:

> *"Methodologically, the essential difference between constrained and unconstrained problems is not determined by the problem itself, but by the algorithm we adopt to solve the problem."*

In problem (UEO), formalistically, x is still subject to the real Euclidean space constraint $\mathbb{R}^n$, i.e., it must be an n-dimensional vector of real numbers rather than a vector of complex numbers. But the iterative formula $x_{k+1} := x_k + t_k d_k$ in the line search framework of Algorithm 1 never breaks this, thus Algorithm 1 is free, and feasibility (with respective to $x \in \mathbb{R}^n$) is guaranteed for any direction d_k or step size t_k so that we have neglected to discuss feasibility at all. Finally, minimization of the value of objective function is the only thing needed to consider.

On the other hand, in dealing with constrained optimization, if we can guarantee that each iteration point of the algorithm employed satisfies the constraints, then this is no different from (UEO). In particular, let us consider the general form as (EO) where feasible region is denoted by S. Note that in (CEO), $S = \{x \in \mathbb{R}^n : g_i(x) = 0, i = 1, 2, \ldots, m \text{ and } h_j(x) \leq 0, j = 1, 2, \ldots, l\}$. Assume that there is an algorithm that generates a sequence $\{x_k\} \subset S$ by using some updating formula:

$$x_{k+1} := \mathrm{UPDATE}(x_k, d_k, t_k).$$

Here, UPDATE is a map such that its output $x_{k+1} \in S$ for any current point $x_k \in S$, any direction d_k and step size $t_k > 0$. In this case, as in unconstrained optimization, only the minimization of the value of objective function needs to be considered, for which we must choose the appropriate direction and step size. In this way, the formalistic constrained problem also becomes unconstrained one. In fact, Riemannian optimization, the central subject of this thesis, realizes the above desires when the feasible region S have a manifold structure.

1.1.2 Riemannian Optimization

Riemannian optimization is a specialized field of optimization focusing on minimizing a function defined over Riemannian manifolds. It extends traditional optimization algorithms by taking the geometry structure of the manifold into account. By definition, a manifold $\mathcal{M}$ is a set that locally resembles a Euclidean space within a neighborhood of each point, but globally may exhibit different geometric properties, usually non-Euclidean (e.g., sphere, hyperboloid). See Table 1.1 for a list of some manifolds $\mathcal{M}$. A typical Riemannian optimization is given as follows.

Problem 3 (Unconstrained Riemannian Optimization (URO))**.** Given a function $f : \mathcal{M} \to \mathbb{R}$, we consider

$$\begin{aligned} &\min\ f(x) \\ &\text{s.t. } x \in \mathcal{M}. \end{aligned} \tag{URO}$$

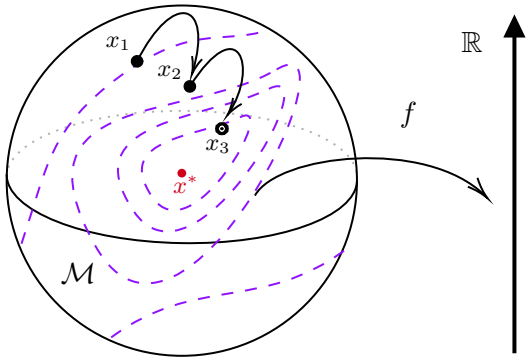


Fig. 1.1 Iteration of optimizing a real-valued function f defined on sphere manifold.

It is well known that Euclidean spaces are the most trivial Riemannian manifolds, and thus (URO) is a generalized form of the problem (UEO). Naturally, the line search framework also applies to (URO), see Fig. 1.1 and Algorithm 2, where several notations come from Riemannian geometry, such as tangent spaces $T_{x_k}\mathcal{M}$ and retraction (maps) $\mathrm{R}_{x_k} : T_{x_k}\mathcal{M} \to \mathcal{M}$. Due to the lack of linearity, update formulas on manifolds cannot generally be defined by addition. Therefore, we introduce the concept of retraction. Riemannian geometry, as detailed in Chapter 2, provides the fundamental concepts and tools necessary to develop and analyze optimization algorithms on Riemannian manifolds.

Algorithm 2: General Line Search Method for (URO)

Input: An objective function f defined on $\mathcal{M}$, an initial point $x_0 \in \mathcal{M}$ and a retraction R on $\mathcal{M}$.

Output: Sequence $\{x_k\} \subset \mathcal{M}$.

Set $k \to 0$;

while *stopping criterion not satisfied* **do**

1. Compute a search direction $d_k \in T_{x_k}\mathcal{M}$;
2. Compute a step size $t_k > 0$;
3. Compute the next point as $x_{k+1} := \mathrm{R}_{x_k}(t_k d_k)$;
4. $k \to k + 1$;

end

Note that the line search method is an iterative framework based on local information of objective function at current point (e.g., the first-order derivative — gradient, or second-order derivative — Hessian), and in particular it utilizes this local information to choose the appropriate direction toward next better point. On the other hand, for every point x on the manifold, there is a neighborhood that is homeomorphic to an open subset of the Euclidean space, and so this neighborhood of x can be considered to have a differential structure, thus we can exploit the local information of the objective function defined on the manifold. This exactly fulfills the request for line search method.

The unconstrained Riemannian optimization has grown considerably in the last 20 years. In particular, well-known methods in the Euclidean setting, such as steepest descent, conjugate gradient, (quasi-)Newton and trust region, have been extended to the Riemannian setting [5, 101, 34, 165]. Riemannian optimization has many advantages in comparison to Euclidean optimization, such as the ability to transform constrained

problems into unconstrained ones (see Table 1.1), the use of the geometric structure of the feasible region, and a convergence theory similar to that of Euclidean optimization.

1.1.3 History of Riemannian Optimization

The concept of Riemannian optimization can be traced back to the work of Luenberger in 1970s. [130, 131] introduced the idea of performing line searches along geodesics (i.e., a generalized concept of straight lines on manifolds) when geodesics are computationally feasible. [79] formally discussed minimizing differentiable functions over differential manifolds, introducing the steepest descent, Newton method, and quasi-Newton methods, all along geodesics, specifically on submanifolds of $\mathbb{R}^n$. [94] explored the optimization problem for dynamical systems on Riemannian manifolds. [72] developed the Newton and conjugate gradient algorithms on the Grassmann and Stiefel manifolds. Importantly, the work of [94, 72] strongly contributed to numerical practice of Riemannian optimization. In general, an approximation of geodesic is sufficient to ensure the desired convergence properties and is less computationally expensive than geodesic. In 2002, [6] defined the concept of retraction (i.e., R_{x_k} in Algorithm 2) to replace the role of geodesic in line search. This had greatly facilitated the numerical performance of Riemannian optimization in practical applications.

Since 2008, Riemannian optimization has been greatly developed, become practicable and convenient, and gradually applied to various engineering fields. [5] summarized many retraction-based algorithms on Riemannian manifolds and is the first modern monograph about Riemannian optimization. The authors suggested that parallel transport in classical Riemannian geometry can be approximated by vector transport without affecting the convergence properties. [2, 11] proposed the trust-region methods on Riemannian manifolds. [153] extended the classical Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm on Riemannian manifolds, along with convergence results. [157] provided an alternative approach to BFGS on Riemannian manifolds, which focused on infinite-dimensional manifolds. [105] proposed a Riemannian BFGS method without differentiated retraction. [201] proposed an optimization algorithm on Stiefel manifold based on the Cayley transform. [107] discussed a Broyden class of quasi-Newton methods for Riemannian optimization. [168, 163, 220, 166] investigated the Riemannian conjugate gradient method. [29, 215, 169] investigated the stochastic algorithm on Riemannian manifolds. [36] discussed the global convergence rate of Riemannian optimization. [75, 14] explored the multi-objective optimization algorithms on Riemannian manifolds. [51, 111] discussed the Riemannian proximal gradient methods. Since Riemannian optimization relies on the chosen metric, [85, 84] proposed Riemannian preconditioned techniques to accelerate algorithms. They enable the manifolds to be endowed with a preconditioning metric.

It can be seen that the Riemannian optimization theory has become a very active research field in recent years, which has had a significant impact on the design and research of modern optimization algorithms.

1.1.4 Applications of Riemannian Optimization

Riemannian geometry is a generalization of Euclidean geometry. It includes very rich nonlinear spaces such as hyperbolic spaces [44, 196, 195], the special Euclidean and orthogonal groups [176, 80], positive definite matrices [21, 189], Grassmann manifolds of subspaces [72, 12, 34], and Stiefel manifolds of

orthogonal matrices [72, 5], among others. Table 1.1 provides a glance of the some (not all) available manifolds in the "Manopt" toolbox.

Riemannian optimization is widely used in various fields such as eigenvalue decomposition [38, 5]; singular value decomposition [167, 162]; low-rank matrix completion [35, 198, 138, 43, 83]; low-rank tensor completion [123, 118, 186]; joint diagonalization [206, 3, 190, 171]; joint singular value decomposition [168]; nonlinear or inverse eigenvalue problems [217, 212, 211]; p-harmonic flow [87]; quantum technologies [129]; learning taxonomy embeddings [144, 145]; neural networks [112, 113, 82, 46, 143, 152, 154]; shape analysis [185, 108]; density estimation [98, 93]; optimal transport [55, 8, 177, 139, 92]; cryo-electron microscopy (cryo-EM) [63]; control theory [170, 173, 172]; max-cut problem [140, 101]; Bose–Einstein condensates [203]; phase retrieval problem [109, 110, 65]; multi-antenna channel communications [218, 86]; compressed modes (CMs) problem [151]; dictionary learning [54, 187]; Gaussian mixture models [97]; elliptical distributions in statistics [184, 216].

For a survey of Riemannian optimization, see [101]. Below are brief summaries of some latest monographs and software libraries of Riemannian Optimization for the reader's convenience.

Monographs of Riemannian Optimization

- An Introduction to Optimization on Smooth Manifolds [34]
- Riemannian Optimization and Its Applications [165]
- Optimization Algorithms on Matrix Manifolds [5]
- Convex Functions and Optimization Methods on Riemannian Manifolds [194]
- Multivariate Data Analysis on Matrix Manifolds [192]
- Population-Based Optimization on Riemannian Manifolds [78]

Libraries of General-purpose Riemannian Optimization Toolboxes These libraries are used to implement Riemannian optimization algorithms as general solvers:

- Manopt [37] in Matlab (the most comprehensive toolbox)
  https://www.manopt.org/
- Pymanopt [191] in Python
  https://pymanopt.org/
- ROPTLIB [106] in C++
  https://www.math.fsu.edu/~whuang2/Indices/index_ROPTLIB.html
- ManifoldOptim [133] in R (a R wrapper of ROPTLIB)
  https://cran.r-project.org/web/packages/ManifoldOptim/index.html
- Manopt.jl [16] in Julia
  https://manoptjl.org/

Libraries of Riemannian Packages for Various Goals These libraries are used to implement various specific tasks based on Riemannian optimization:

- Geoopt [120] is a Python library bringing Riemannian optimization tools to PyTorch.
  https://geoopt.readthedocs.io/en/latest/index.html
- McTorch [135] is also a Python library bringing Riemannian optimization tools to PyTorch.
  https://github.com/mctorch/mctorch

Table 1.1 Collection of some available manifolds in Manopt.

Name of Manifold	**Mathematical Formulation**
(Complex) Euclidean Space	$\mathbb{R}^{m\times n}, \mathbb{C}^{m\times n}$
Symmetric Matrices	$\left\{X \in \mathbb{R}^{n\times n} : X = X^T\right\}$
Skew-Symmetric Matrices	$\left\{X \in \mathbb{R}^{n\times n} : X + X^T = 0\right\}$
Centered Matrices	$\{X \in \mathbb{R}^{m\times n} : X\mathbf{1}_n = 0_m\}$
Sphere	$\{X \in \mathbb{R}^{m\times n} : \|X\|_{\mathrm{F}} = 1\}$
Symmetric Sphere	$\left\{X \in \mathbb{R}^{n\times n} : \|X\|_{\mathrm{F}} = 1, X = X^T\right\}$
Complex Sphere	$\{X \in \mathbb{C}^{m\times n} : \|X\|_{\mathrm{F}} = 1\}$
Oblique Manifold	$\left\{X \in \mathbb{R}^{m\times n} : \|X_{:,1}\|_{\mathrm{F}} = \cdots = \|X_{:,n}\|_{\mathrm{F}} = 1\right\}$
Complex Oblique Manifold	$\left\{X \in \mathbb{C}^{m\times n} : \|X_{:,1}\|_{\mathrm{F}} = \cdots = \|X_{:,n}\|_{\mathrm{F}} = 1\right\}$
Complex Circle	$\{z \in \mathbb{C}^n : \|z_1\| = \cdots = \|z_n\| = 1\}$
Phase of Real DFT	$\left\{z \in \mathbb{C}^n : \|z_k\| = 1, z_{1+ \bmod (k,n)} = \bar{z}_{1+ \bmod (n-k,n)}, \forall k\right\}$
Stiefel Manifold	$\left\{X \in \mathbb{R}^{n\times p} : X^T X = I\right\}$
Complex Stiefel Manifold	$\{X \in \mathbb{C}^{n\times p} : X^* X = I\}$
Generalized Stiefel Manifold	$\left\{X \in \mathbb{R}^{n\times p} : X^T B X = I\right\}$ for some $B \succ 0$
Grassmann Manifold	$\left\{\mathrm{span}(X) : X \in \mathbb{R}^{n\times p}, X^T X = I\right\}$
Complex Grassmann Manifold	$\{\mathrm{span}(X) : X \in \mathbb{C}^{n\times p}, X^* X = I\}$
Generalized Grassmann Manifold	$\left\{\mathrm{span}(X) : X \in \mathbb{R}^{n\times p}, X^T B X = I\right\}$ for some $B \succ 0$
Rotation Group	$\left\{R \in \mathbb{R}^{n\times n} : R^T R = I, \det(R) = 1\right\}$
Special Euclidean Group	$\left\{(R, t) \in \mathbb{R}^{n\times n} \times \mathbb{R}^n : R^T R = I, \det(R) = 1\right\}$
Unitary Matrices	$\{U \in \mathbb{C}^{n\times n} : U^* U = I_n\}$
Hyperbolic manifold	$\left\{x \in \mathbb{R}^{n+1} : x_0^2 = x_1^2 + \cdots + x_n^2 + 1\right\}$ with Minkowski metric
Fixed-Rank Manifold	$\{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) = k\}$
Fixed-Rank Tensor, Tucker	Tensors of fixed multilinear rank in Tucker format
Strictly Positive Matrices	$\{X \in \mathbb{R}^{m\times n} : X_{ij} > 0, \forall i, j\}$
Symmetric Positive Definite Matrices	$\left\{X \in \mathbb{R}^{n\times n} : X = X^T, X \succ 0\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^T \succeq 0, \mathrm{rank}(X) = k\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^T \succeq 0, \mathrm{rank}(X) = k, \mathrm{diag}(X) = \mathbf{1}\right\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X = X^T \succeq 0, \mathrm{rank}(X) = k, \mathrm{trace}(X) = 1\right\}$
Multinomial manifold	$\{X \in \mathbb{R}^{m\times n} : X_{ij} > 0, \forall i, j$ and $X\mathbf{1}_n = \mathbf{1}_m\}$
-	$\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0, \forall i, j$ and $X\mathbf{1}_n = \mathbf{1}_n, X^T\mathbf{1}_n = \mathbf{1}_n\}$
-	$\left\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0, \forall i, j \text{ and } X\mathbf{1}_n = \mathbf{1}_n, X = X^T\right\}$
Positive Definite Simplex	$\{(X_1, 2, \ldots, x_k) \in \mathbb{R}^{n\times n} : X_i \succ 0, \forall i$ and $X_1 + \cdots + x_k = I_n\}$
Complex Positive Definite Simplex	$\{(X_1, 2, \ldots, x_k) \in \mathbb{C}^{n\times n} : X_i \succ 0, \forall i$ and $X_1 + \cdots + x_k = I_n\}$
Sparse Matrices of Fixed Sparsity Pattern	$\{X \in \mathbb{R}^{m\times n} : X_{ij} = 0 \Leftrightarrow A_{ij} = 0\}$
Constant Manifold (singleton)	$\{A\}$

- TensorFlow RiemOpt [180] is a library for Riemannian optimization in TensorFlow.
  https://github.com/master/tensorflow-riemopt
- Rieoptax [197] is a library for Riemannian Optimization in JAX.
  https://github.com/SaitejaUtpala/rieoptax
- CDOpt [204] is a Python toolbox for optimization on Riemannian manifolds with support for deep learning.
  https://cdopt.github.io/md_files/intro.html
- QGOpt [129] is an extension of TensorFlow optimizers on Riemannian manifolds that often arise in quantum mechanics.
  https://qgopt.readthedocs.io/en/latest/#
- Geomstats [137] is a Python package for computations and statistics on manifolds.
  https://geomstats.github.io/

1.2 Nonsmooth Riemannian Optimization

1.2.1 Introduction to Nonsmooth Riemannian Optimization

Nonsmooth Riemannian Optimization (NRO) involves minimizing a nonsmooth objective function defined on the Riemannian manifold. The optimization techniques in previous section mainly deal with smooth objective functions, and therefore new methods need to be developed to effectively deal with the nonsmooth case.

Problem 4 (Nonsmooth Riemannian Optimization (NRO))**.** Given a nonsmooth function $f\colon \mathcal{M} \to \mathbb{R}$, we consider

$$\begin{aligned} &\min\ f(x) \\ &\text{s.t. } x \in \mathcal{M} \end{aligned} \tag{NRO}$$

There have been some derivative-free optimization techniques specifically for manifolds. The direct search method is a function minimization algorithm that uses only evaluations of the function $f(x)$ itself. [66] extended three popular direct search methods, namely, the Nelder-Mead simplex algorithm, the Mesh-Adapted Direct Search (MADS) algorithm, and frame-based methods, to Riemannian manifolds. In [47], Powell's derivative-free optimization method is extended to Oblique manifolds to recover quasi-correlated sources by minimizing the contrast function. [30] proposed to adapt the particle swarm optimization algorithm on Grassmann manifolds to find the best low multilinear rank approximation for a given tensor.

One type of problems (NRO) we are interested in has a locally Lipschitz objective function, and thus almost everywhere differentiable. A larger number of algorithms have been developed with this locally Lipschitz assumption. [91] presented an ε-subgradient algorithm on Riemannian manifolds. [90] proposed the nonsmooth trust region algorithms on manifolds. [100, 42] generalized a gradient sampling algorithm to the Riemannian setting. [99] generalized the Wolfe conditions for nonsmooth functions on manifolds and extended the nonsmooth BFGS algorithm. [13, 75] investigated the iteration complexity of many Riemannian subgradient algorithms. [124] presented a splitting method for orthogonality constrained problems. [122] proposed the Manifold Alternating Directions Method of Multipliers (MADMM), an extension of the classical ADMM scheme for manifold-constrained nonsmooth optimization problems.

[51] proposed a Manifold Proximal Gradient (ManPG) method. [200] proposed a accelerated version of the proximal alternating maximization method.

1.2.2 Applications of Nonsmooth Riemannian Optimization

[4] collected many important applications of (NRO). (NRO) may be required for any application that already is modeled using (smooth) Riemannian optimization. For example, we may add some nonsmooth regularization terms to the objective function, or add additional non-manifold constraints as exact penalty terms to the objective function as well (see (1.2) in next section).

We next provide two concrete application problems below, which will appear in the numerical experiments in the latter Chapter 3 to test the efficiency of our proposal — Riemannian Smoothing Method (RSM).

Finding the Sparsest Vector (FSV)

This problem is to find the sparsest vector in an n-dimensional linear subspace $W \subset \mathbb{R}^m$; it has applications in robust subspace recovery, dictionary learning, and many other problems in machine learning and signal processing [155, 156]. Let $Q \in \mathbb{R}^{m\times n}$ denote a matrix whose columns form an orthonormal basis of W: this problem can be formulated as

$$\min_{x\in \mathrm{Sp}^{n-1}} \|Qx\|_0,$$

where $\mathrm{Sp}^{n-1} := \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$ is the sphere manifold, and $\|z\|_0$ counts the number of nonzero components of vector z. Because this discontinuous objective function is unwieldy, in the literature, one instead focuses on solving the ℓ_1 norm relaxation given below:

$$\min_{x\in \mathrm{Sp}^{n-1}} \|Qx\|_1, \tag{FSV}$$

where $\|z\|_1 := \sum_i |z_i|$ is the ℓ_1 norm of the vector z.

Robust Low-Rank Matrix Completion

Low-rank matrix completion [198, 138] consists of recovering a rank r matrix M of size $m \times n$ from only a fraction of its entries with $r \ll \min\{m, n\}$. The situation in robust low-rank matrix completion [43] is one where only a few observed entries, called outliers, are perturbed, i.e., $M = M_0 + S$, where M_0 is the unperturbed original data matrix of rank r and S is a sparse matrix. This is a case of adding non-Gaussian noise for which the traditional ℓ_2 minimization model,

$$\min_{X\in \mathrm{Fr}(m,n,r)} \|P_\Omega(X - M)\|_2^2$$

is not well suited to recovery of M_0. Here, $\mathrm{Fr}(m, n, r) := \{X \in \mathbb{R}^{m\times n} : \mathrm{rank}(X) = r\}$ is the fixed rank manifold, Ω denotes the set of indices of observed entries, and $P_\Omega : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$ is the orthogonal projector onto Ω, defined as $Z \mapsto P_\Omega(Z)$ with $P_\Omega(Z)_{ij} = Z_{ij}$ if $(i, j) \in \Omega$ and 0 otherwise. In [43], the

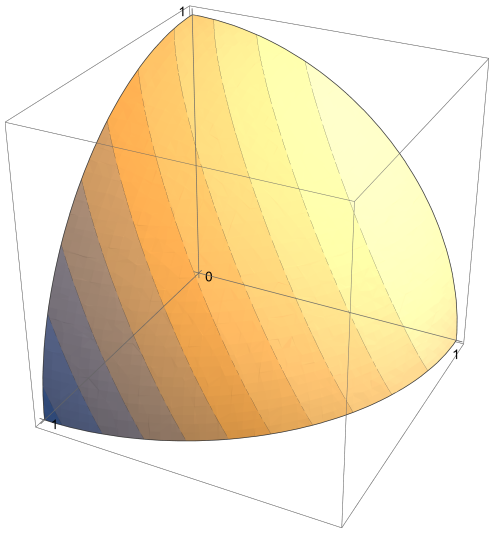


Fig. 1.2 Illustration of nonnegative constrains on sphere manifold.

authors try to solve

$$\min_{X \in \mathrm{Fr}(m,n,r)} \|P_\Omega(X - M)\|_1 , \tag{RMC}$$

because the sparsity-inducing property of the ℓ_1 norm leads one to expect exact recovery when the noise consists of just a few outliers.

1.3 Constrained Riemannian Optimization

1.3.1 Introduction to Constrained Riemannian Optimization

In most cases, the term "Riemannian optimization" refers to (URO) — Unconstrained Riemannian Optimization, which has become more and more developed and sophisticated over the last two decades. However, some limitations of this model have emerged in practical applications.

One of which is that (URO) requires the entire feasible region to form exactly one manifold. Unfortunately, in many practical cases, only a fraction of the constraints can form a manifold, leaving us with additional constraints that must be addressed. For example, see Fig. 1.2. Furthermore, even if we manage to create a new manifold with these additional constraints (meaning not among the Table 1.1, so there is no off-the-shelf software package), finding the necessary geometric tools (e.g., tangent space, retraction, vector transport) is a challenging task, as it requires a strong knowledge of the differential geometry, which can discourage most users.

To address these challenges, we consider replacing the (URO) with a model called the Constrained Riemannian Optimization (CRO), which extends the (URO) by adding equality and inequality constrained conditions defined on the manifold itself. Just as it is natural for us to consider constraints underlying Euclidean spaces, we can also recognize those constraints on manifolds. After all, Euclidean space is a trivial manifold.

Problem 5 (Constrained Riemannian Optimization (CRO))**.** Given a function $f\colon \mathcal{M} \to \mathbb{R}$, we consider

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0,\ i = 1, 2, \ldots, m, \\ & h_j(x) = 0,\ j = 1, 2, \ldots, l, \\ & x \in \mathcal{M}, \end{aligned} \tag{CRO}$$

where $g_i\colon \mathcal{M} \to \mathbb{R}$ and $h_j\colon \mathcal{M} \to \mathbb{R}$ are functions over manifold $\mathcal{M}$.

Research on the constrained Riemannian optimization is still in its infancy. The earliest studies go back to the Riemannian optimality conditions of (CRO). [209] extended the Karush Kuhn Tucker (KKT) conditions and the second-order necessary and sufficient conditions to (CRO). [17] considered more Constraint Qualifications (CQs) on manifolds. [207] proposed sequential optimality conditions, also called approximate KKT conditions, in the Riemannian case.

In 2020, [128] were the first to develop practical algorithms for (CRO). They proposed the Riemannian version of augmented Lagrangian method and exact penalty method. Riemannian Augmented Lagrangian Method (RALM) [128] relies on the augmented Lagrangian function:

$$\mathcal{L}_\rho(x, \lambda, \gamma) := f(x) + \frac{\rho}{2}\left(\sum_j \left(h_j(x) + \frac{\gamma_j}{\rho}\right)^2 + \sum_i \max\left\{0, \frac{\lambda_i}{\rho} + g_i(x)\right\}^2\right), \tag{1.1}$$

where $x \in \mathcal{M}$, $\rho > 0$ is a penalty parameter and $\gamma \in \mathbb{R}^l, \lambda \in \mathbb{R}^m, \lambda \geq 0$ are Lagrangian multipliers. RALM alternates between updating x and updating (λ, γ, ρ). To update x, any algorithm for (URO) may be adopted to minimize (1.1) with (λ, γ) fixed, namely, compute new iterate x_{k+1} as an approximate solution to $\min_{x\in\mathcal{M}} \mathcal{L}_{\rho_k}(x, \lambda_k, \gamma_k)$. At this new x_{k+1}, $\lambda_{k+1}, \gamma_{k+1}$ and ρ_{k+1} will be updated; then the process is repeated. Soon after, [207] improved RALM in order to obtain an optimal solution without satisfying CQs. Riemannian Exact Penalty Method (REPM) [128] tries to solve following subproblem at each iteration:

$$\min_{x\in\mathcal{M}}\ f(x) + \rho\left(\sum_i \max\{0, g_i(x)\} + \sum_j |h_j(x)|\right) \tag{1.2}$$

However, the above objective function is nonsmooth, then they use smoothing techniques to convert it to a smooth objective function, and again, any algorithm for (URO) can be adopted. They essentially use the Riemannian smoothing method as proposed in Chapter 3 of this thesis. [174, 148] proposed the Riemannian Sequential Quadratic Programming (RSQP) method. RSQP attempts to solve following quadratic programming on tangent space at each iteration x_k:

$$\begin{aligned} \min_{\Delta x_k \in T_{x_k}\mathcal{M}} \quad & \tfrac{1}{2}\left\langle B_k\left[\Delta x_k\right], \Delta x_k\right\rangle + \left\langle \operatorname{grad} f(x_k), \Delta x_k\right\rangle \\ \text{s.t.} \quad & g_i\left(x_k\right) + \left\langle \operatorname{grad} g_i\left(x_k\right), \Delta x_k\right\rangle \leq 0,\ i = 1, 2, \ldots, m, \\ & h_j\left(x_k\right) + \left\langle \operatorname{grad} h_j\left(x_k\right), \Delta x_k\right\rangle = 0,\ j = 1, 2, \ldots, l. \end{aligned} \tag{1.3}$$

The mathematical notations above can be referred to in the next chapter on Riemannian geometry. Here, $B_k : T_{x_k}\mathcal{M} \to T_{x_k}\mathcal{M}$ is a modified Hessian of the Lagrangian function and is a symmetric positive definite linear operator. [148] first convert (1.3) into Euclidean form and then solve. [103] proposed a

heuristic algorithm for the submanifold with nonnegative constraints by imitating the techniques from Riemannian optimization.

We should emphasize that (CRO) retains the advantages of (URO) and successfully solves the previously mentioned issues, i.e., (CRO) only requires some of the constraints to be manifolds, not all. Moreover, the algorithm design for (CRO) can still use the existing manifold toolbox. For example, the RALM and REPM algorithms introduced previously are based on Manopt's various solvers (actually they are wrappers of Manopt) and do not require any new geometric tools to be developed on $\mathcal{M}$.

1.3.2 Applications of Constrained Riemannian Optimization

Constrained Riemannian optimization feature naturally in applications such as orthogonal nonnegative matrix factorization [125, 210]; subproblem of k-indicators model for data clustering [49, 116]; minimum balanced cut for graph bisection [128]; nonnegative principal component analysis [141]; k-means via low-rank SDP [45]. (CRO) may be required for any application that already is modeled using unconstrained Riemannian optimization. Because it is common to add some new constraints to an existing (URO) model, such as non-negative constraints.

We next provide two concrete application problems below, which will appear in the numerical experiments in the latter Chapter 6 to test the efficiency of our proposal — Riemannian Interior Point Methods (RIPM).

Nonnegative Low-Rank Matrix (NLRM) Approximation

Recently, [182] proposed the NLRM approximation method, which is different from the classical Nonnegative Matrix Factorization (NMF) method, i.e., $\min_{B,C\geq 0} \|A - BC\|_{\mathrm{F}}^2$ for $B \in \mathbb{R}^{m\times r}$, $C \in \mathbb{R}^{r\times n}$. Mathematically, NLRM aims to solve

$$\min_{X\in \mathrm{Fr}(m,n,r)} \|A - X\|_{\mathrm{F}}^2 \quad \text{s.t. } X \geq 0. \tag{NLRM}$$

Clearly, NLRM can obtain a better nonnegative low-rank matrix approximation than that of NMF.

Projection onto Nonnegative Stiefel Manifold

Given $C \in \mathbb{R}^{n\times k}$, we compute its projection onto the nonnegative part of the Stiefel manifold. The Stiefel manifold, denoted as $\mathrm{St}(n, k)$, is the set of $n \times k$ matrices X with orthonormal columns. That is, $X^TX = I_k$ where I_k is the $k \times k$ identity matrix. Consider the distance measured by the Frobenius norm, we have $\|C - X\|_{\mathrm{F}}^2 = \mathrm{trace}((C - X)^T(C - X))$. To minimize this distance, we expand:

$$\|C - X\|_{\mathrm{F}}^2 = \mathrm{trace}(C^TC) - 2\,\mathrm{trace}(X^TC) + \mathrm{trace}(X^TX).$$

Since X lies on the Stiefel manifold, $\mathrm{trace}(X^TX) = \mathrm{trace}(I_k) = k$. The term $\mathrm{trace}(C^TC)$ is constant with respect to X. Therefore, our problem can be equivalently formulated as

$$\min_{X\in \mathrm{St}(n,k)} -2\,\mathrm{trace}(X^TC) \quad \text{s.t. } X \geq 0, \tag{Model\_St}$$

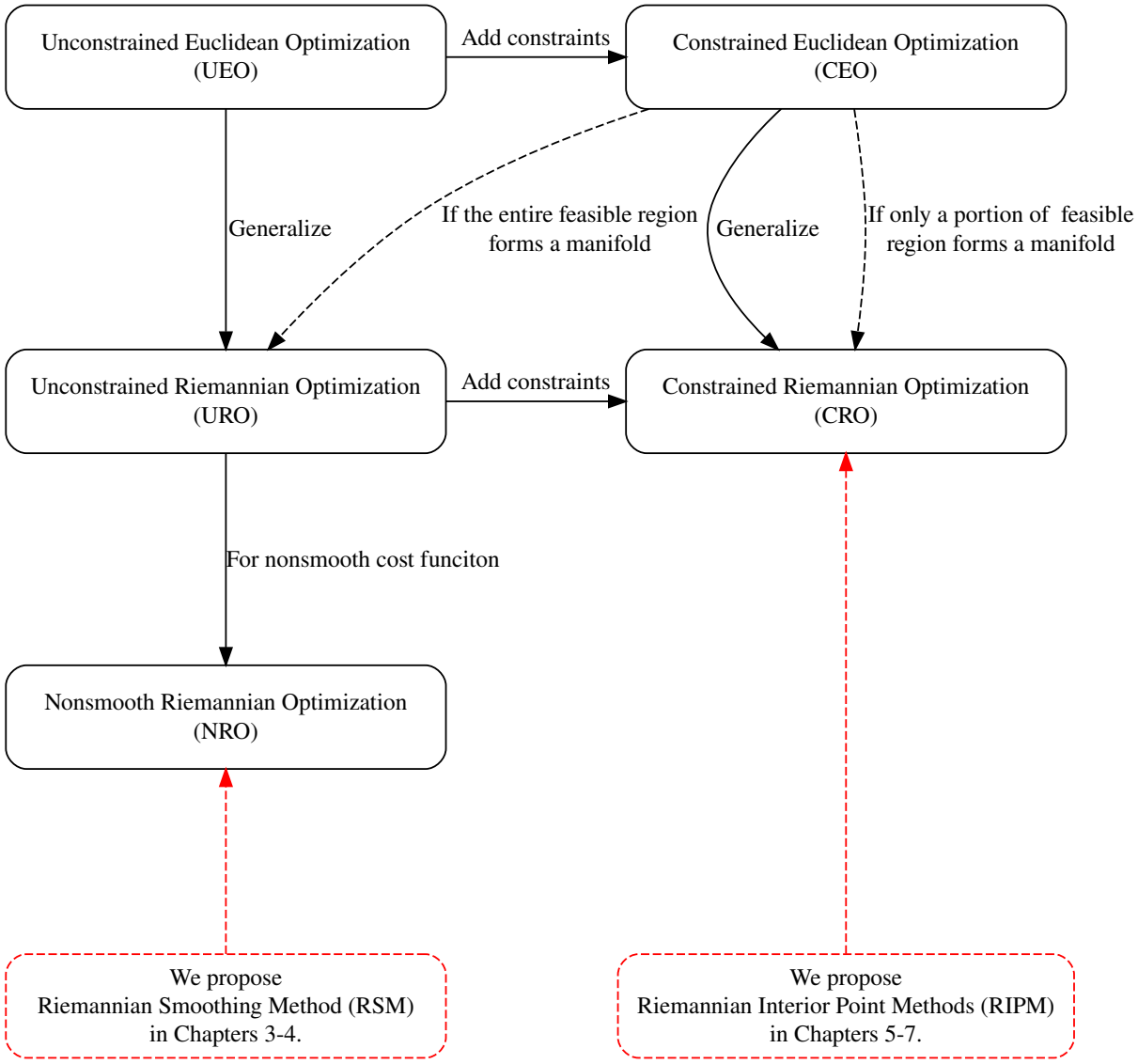


Fig. 1.3 Relationship of several types of Euclidean optimization and Riemannian optimization introduced in this chapter.

which has a linear objective function. In [116], it is shown that (Model_St) can be reformulated into

$$\min_{X\in \mathrm{Ob}(n,k)} -2\,\mathrm{trace}(X^T C) \quad \text{s.t. } X \geq 0, \text{ and } \|XV\|_{\mathrm{F}} = 1, \tag{Model_Ob}$$

where the positive integer p and matrix $V \in \mathbb{R}^{k\times p}$ can be arbitrary as long as $\|V\|_{\mathrm{F}} = 1$ and VV^T is entrywise positive. In particular, the feasible region of (Model_St) and (Model_Ob) are identical.

1.4 Summary and Thesis Structure

Fig. 1.3 displays the relationship between all the various types of Euclidean optimization (UEO, CEO) and Riemannian optimization (URO, NRO, CRO) mentioned in the previous three sections. There is a close relationship between them. As shown in Fig. 1.3, the main contribution of this thesis is to propose

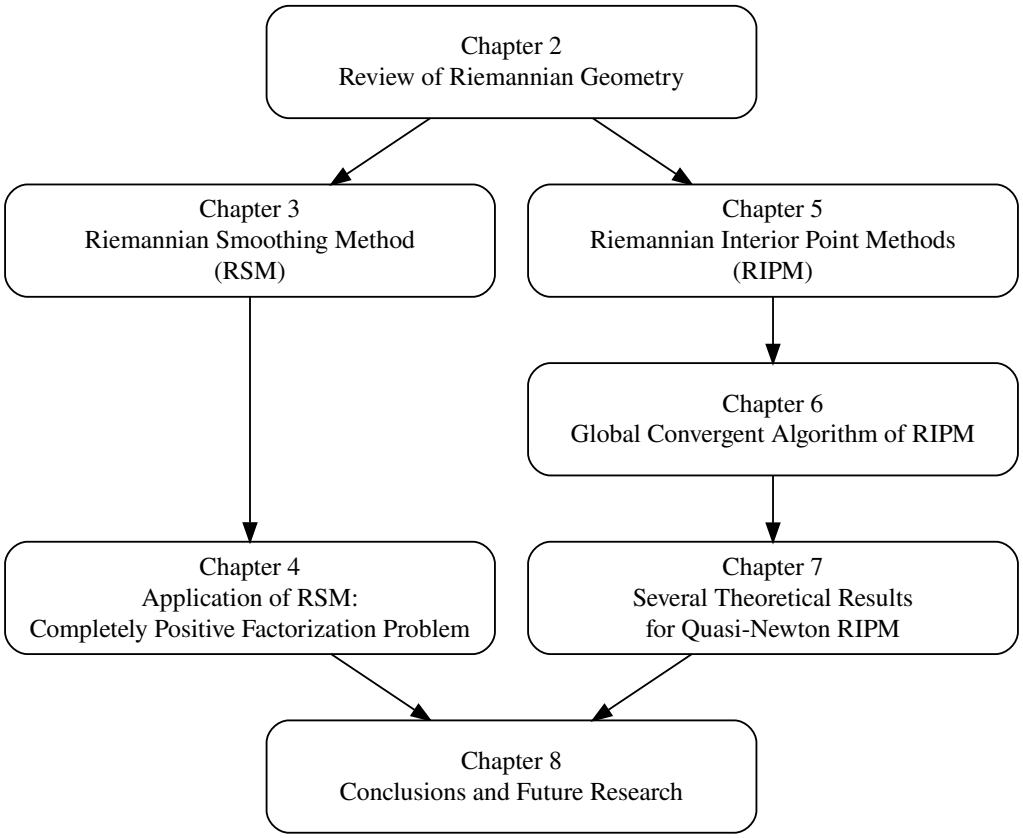


Fig. 1.4 The rest of the thesis begins with a review of Riemannian geometry providing the various tools for optimization, followed by two proposed algorithms that are independent of each other: the RSM and the RIPM. It ends with conclusions and future work.

novel algorithms for two of these types of Riemannian optimization. Fig. 1.4 summarizes the structure and reading order of the rest of this thesis. The content of this thesis is divided into three main Parts, whose corresponding chapters and outlines are shown below:

Part I: Preliminary

- Chapter 1 (this chapter) provides an overview of several different types of Riemannian optimization problems and gives the structure of this thesis.
- Chapter 2 introduces tools and useful conclusions in Riemannian geometry that we will use later. Readers familiar with Riemannian geometry can skip this chapter.

Part II: Proposal I - Riemannian Smoothing Method (RSM)

- Chapter 3 presents the general framework of the Riemannian smoothing method for nonsmooth Riemannian optimization.
- Chapter 4 deals with the application of Riemannian smoothing method to the problem of completely positive matrix factorization, an open problem in conic optimization theory.

Part III: Proposal II - Riemannian Interior Point Methods (RIPM)

- Chapter 5 presents the Riemannian interior point method for constrained Riemannian optimization. We will discuss the details of all theoretical generalizations and a prototype algorithm and its local convergence.
- Chapter 6 gives a globally convergent Riemannian interior point algorithm and proves its global convergence. In addition, numerical experiments verify its effectiveness.
- Chapter 7 explores the Riemannian interior point method using the quasi-Newton method.
- Chapter 8 gives conclusions and future work of this thesis.

Chapter 2

Review of Riemannian Geometry

Part	Section
Part 1. Basic tools	2.1 Linear Algebra 2.2 Smooth Manifold 2.3 Tangent Vectors 2.4 Derivatives of Maps 2.5 Embedded Submanifold 2.6 Vector Fields 2.7 Retractions 2.8 Riemannian Manifolds 2.9 Connections, Covariant Derivatives and Hessians 2.10 Induced Covariant Derivative and Geodesics 2.11 Exponential Map 2.12 Metric Space
Part 2. Advanced tools	2.13 Parallel Transport 2.14 Vector Transports 2.15 Totally Retractive Neighborhood 2.16 Lipschitz Continuity with Respect to a Vector Transport
Part 3. Auxiliary results	2.17 Local Errors of Retractions and Vector Transports 2.18 Fundamental Theorem of Calculus in Riemannian Case 2.19 Some Lemmas on Newton method 2.20 Geometry Tools of Product Manifold $\mathcal{N}$ 2.21 Note and References

This chapter introduces the basic tools of Riemannian geometry, as well as useful results that are indispensable to our subsequent chapters. For those readers already familiar with Riemannian geometry, please continue to the next chapter. Table 2.1 in next page illustrates a quick comparison of the various concepts/tools of Riemannian manifolds and familiar Euclidean space (typical, $\mathbb{R}^n$). The reader may check this table from time to time while reading this chapter. Keep in mind that manifolds are generalizations of Euclidean space.

Table 2.1 Comparison between an n-dimensional Riemannian manifold $\mathcal{M}$ and Euclidean Space $\mathbb{R}^n$.

Concepts	**Riemannian manifold $\mathcal{M}$**	**Euclidean $\mathbb{R}^n$**
Linear Structure	Generally, no	Yes
Dimension	n (see Definition 1)	n
Tangent Space	$T_x\mathcal{M}$, an n-dimensional vector space, depends on x (see Definition 7 and Proposition 1)	$\mathbb{R}^n$ itself, does not depend on x
Smooth Map	$F\colon \mathcal{M} \to \mathcal{N}$ (see Definition 5)	$F\colon \mathbb{R}^n \to \mathbb{R}^m$
Differential	$\mathrm{D}F(x)\colon T_x\mathcal{M} \to \mathrm{T}_{F(x)}\mathcal{N}$, a linear operator (see Definition 8)	$\mathrm{D}F(x)\colon \mathbb{R}^n \to \mathbb{R}^m$, and $\mathrm{D}F(x)[u] = \mathbf{J}_F(x)u$ with the Jacobian matrix $\mathbf{J}_F(x) \in \mathbb{R}^{m\times n}$ of F at x
Tangent Bundle	$T\mathcal{M} = \bigcup_{x\in\mathcal{M}} \{(x,\xi) : \xi \in T_x\mathcal{M}\}$ (see Definition 11)	$T\mathbb{R}^n = \mathbb{R}^n \times \mathbb{R}^n$
Vector Field	$V\colon \mathcal{M} \to T\mathcal{M}$ such that $V(x) \in T_x\mathcal{M}$ for all $x \in \mathcal{M}$ (see Definition 12)	$V\colon \mathbb{R}^n \to \mathbb{R}^n$
Motion	Retraction $\mathrm{R}\colon T\mathcal{M} \to \mathcal{M}$ (see Definition 15, 16)	Straight line, $\mathrm{R}(x,\xi) = x + \xi$.
Metric	Riemannian Metric, $\langle\cdot,\cdot\rangle_x : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$, depends on x (see Definition 17)	Euclidean metric, $\langle x, y\rangle = x^T y$, does not depend on x
Scalar Field	$f\colon \mathcal{M} \to \mathbb{R}$	$f\colon \mathbb{R}^n \to \mathbb{R}$
Gradient	$\operatorname{grad} f(x)$, a tangent vector in $T_x\mathcal{M}$ (see Definition 19)	$\operatorname{egrad} f(x)$, a vector in $\mathbb{R}^n$
Gradient Vector Field	$x \mapsto \operatorname{grad} f(x)$, a vector field on $\mathcal{M}$	$x \mapsto \operatorname{egrad} f(x)$, a map form $\mathbb{R}^n$ to $\mathbb{R}^n$
Connection	Levi-Civita or Riemannian connection ∇ (see Theorem 3)	Canonical Euclidean connection (2.27)
Covariant Derivative	For vector field V on $\mathcal{M}$, covariant derivative $\nabla V(x)\colon T_x\mathcal{M} \to T_x\mathcal{M}$, a linear operator (see Definition 22)	For function V from and to $\mathbb{R}^n$, $\nabla V(x)\colon \mathbb{R}^n \to \mathbb{R}^n$, and $\nabla V(x)[u] = \mathbf{J}_V(x)u$ with the Jacobian matrix $\mathbf{J}_V(x) \in \mathbb{R}^{n\times n}$ of V at x
Hessian	For smooth $f\colon \mathcal{M} \to \mathbb{R}$, $\operatorname{Hess} f(x)\colon T_x\mathcal{M} \to T_x\mathcal{M}$ is a symmetric linear operator (see Definition 23)	For smooth $f\colon \mathbb{R}^n \to \mathbb{R}$, $\operatorname{Hess} f(x)\colon \mathbb{R}^n \to \mathbb{R}^n$, and $\operatorname{Hess} f(x)[u] = \mathbf{H}_f(x)u$ with the Hessian matrix $\mathbf{H}_f(x) \in \mathrm{S}(n)$ of f at x
Newton Method	To find singularity of vector field, $V(x) = 0_x \in T_x\mathcal{M}$ (see (Singularity))	To find zero of nonlinear equation, $V(x) = 0$

2.1 Linear Algebra

Let $\mathcal{E}$ and $\mathcal{E}'$ be two finite-dimensional normed vector spaces, equipped with norms induced by inner products $\langle \cdot, \cdot \rangle_{\mathcal{E}}$ and $\langle \cdot, \cdot \rangle_{\mathcal{E}'}$, respectively. Let $\mathcal{A} \colon \mathcal{E} \to \mathcal{E}'$ be a linear operator. The adjoint of $\mathcal{A}$, denoted as $\mathcal{A}^*$, is another linear operator defined from $\mathcal{E}'$ to $\mathcal{E}$, and it is characterized by the following property that for all vectors $v \in \mathcal{E}$ and $u \in \mathcal{E}'$:

$$\langle \mathcal{A}(v), u \rangle_{\mathcal{E}'} = \langle v, \mathcal{A}^*(u) \rangle_{\mathcal{E}}.$$

The operator norm of $\mathcal{A}$, denoted as $\|\mathcal{A}\|$, is the supremum of the norm of $\mathcal{A}(v)$ for all (at most) unit vectors $v \in \mathcal{E}$, where the norm of $\mathcal{A}(v)$ is taken in the vector space $\mathcal{E}'$:

$$\|\mathcal{A}\| := \sup \{ \|\mathcal{A}(v)\|_{\mathcal{E}'} \mid v \in \mathcal{E}, \|v\|_{\mathcal{E}} = 1 \} = \sup \{ \|\mathcal{A}(v)\|_{\mathcal{E}'} \mid v \in \mathcal{E}, \|v\|_{\mathcal{E}} \leq 1 \}. \tag{2.1}$$

For writing convenience, when the operator $\mathcal{A}$ acts on a vector v, and its result may be denoted as $\mathcal{A}v$, omitting the parentheses. Moreover, a linear operator $\mathcal{A} \colon \mathcal{E} \to \mathcal{E}$ is called self-adjoint or symmetric if $\mathcal{A} = \mathcal{A}^*$. In most cases throughout this thesis, the symbol $\mathcal{E}$ and $\mathcal{E}'$ typically denote some tangent spaces (a kind of vector space, see Section 2.3 later). Specifically, our analysis of operators on tangent spaces often draws on our understanding of linear algebra.

Finally, let us focus on orthogonal projector, which is very useful in exploring the geometry tools of submanifold of Euclidean space later on. Let $\mathcal{E}$ be a finite-dimensional vector space equipped with inner product $\langle \cdot, \cdot \rangle$. If $\mathcal{F}$ is a subspace of $\mathcal{E}$, then the orthogonal complement of $\mathcal{F}$ (with respect to $\langle \cdot, \cdot \rangle$), defined by

$$\mathcal{F}^{\perp} := \{ v \in \mathcal{E} : \langle v, u \rangle = 0 \text{ for all } u \in \mathcal{F} \}$$

is also a subspace of $\mathcal{E}$ [9, 6.46 (a)]. Moreover, $\mathcal{F}$ and $\mathcal{F}^{\perp}$ become a direct sum decomposition of $\mathcal{E}$, i.e., each $v \in \mathcal{E}$ can be *uniquely* written in the form $v = u + w$ with $u \in \mathcal{F}$ and $w \in \mathcal{F}^{\perp}$ [9, 6.47]. In this case, we define the orthogonal projection of $\mathcal{E}$ onto $\mathcal{F}$ (with respect to $\langle \cdot, \cdot \rangle$) as the linear operator

$$\mathrm{Proj}_{\mathcal{F}} \colon \mathcal{E} \to \mathcal{F} \subset \mathcal{E} \colon v \mapsto \mathrm{Proj}_{\mathcal{F}}(v) := u.$$

Indeed, $\mathrm{Proj}_{\mathcal{F}}$ is self-adjoint, i.e., $\forall u, v \in \mathcal{E}, \langle u, \mathrm{Proj}_{\mathcal{F}}(v) \rangle = \langle \mathrm{Proj}_{\mathcal{F}}(u), v \rangle$. Refer to [9, 7.32 (a)] or [34, Proposition 3.55] for proofs.

2.2 Smooth Manifold

We begin with a set $\mathcal{M}$ equipped with a topological structure. To minimize an objective function f that is defined on $\mathcal{M}$, understanding how its derivatives behave becomes crucial. This understanding serves as the foundation for developing various optimization techniques, such as the steepest descent method and Newton method. When we extend our consideration beyond the Euclidean space (topological space $\mathcal{M}$ might not be a subset of the Euclidean space) to include broader spaces, defining the derivative of f directly becomes challenging due to the inherent abstraction of $\mathcal{M}$. This leads us to the definition of "manifolds" — a concept that bridges the gap between abstract spaces and differentiable structures. In this

section, we will briefly introduce various manifolds step by step. For a more comprehensive insight into manifolds, see [127].

2.2.1 Topological Manifold

By enhancing the structure of a given topological space $\mathcal{M}$, our goal is to establish a notion of differentiability of functions defined on $\mathcal{M}$. We initiate this process by considering topological manifolds and then proceed to differentiable manifolds in next subsection.

Definition 1 (Topological Manifold)**.** A topological space $\mathcal{M}$ is called an n-dimensional *topological manifold* if it is a second-countable Hausdorff space; and, for any point p in $\mathcal{M}$, there exists an open set $\mathcal{U}$ in $\mathcal{M}$ containing p and a homeomorphism $\varphi\colon \mathcal{U} \to \mathcal{V}$ from $\mathcal{U}$ to an open set $\mathcal{V}$ in $\mathbb{R}^n$. We denote the dimension of $\mathcal{M}$ as $\dim \mathcal{M} = n$.

In Definition 1, the pair $(\mathcal{U}, \varphi)$ is called a *chart* around p. Consequently, p can be represented using an n-tuple of real numbers $\varphi(p) \in \mathbb{R}^n$ through the function φ. If $x_i : \mathcal{U} \to \mathbb{R}$ is used to map a point $p \in \mathcal{U}$ to the i-th component of $\varphi(p)$, the chart $(\mathcal{U}, \varphi)$ can be alternatively denoted as $(\mathcal{U}; x_1, x_2, \ldots, x_n)$, with the notation $\varphi \equiv (x_1, x_2, \ldots, x_n)$.

According to Definition 1, for an n-dimensional topological manifold $\mathcal{M}$, there exists a set of indices denoted as Λ, along with a collection of charts $\{(\mathcal{U}_\lambda, \varphi_\lambda)\}_{\lambda\in\Lambda}$ such that $\mathcal{M} = \bigcup_{\lambda\in\Lambda} \mathcal{U}_\lambda$. This collection of charts, $\{(\mathcal{U}_\lambda, \varphi_\lambda)\}_{\lambda\in\Lambda}$, is called an *atlas* of $\mathcal{M}$.

2.2.2 Differentiable Manifold

For any point p in $\mathcal{M}$ and a chart $(\mathcal{U}, \varphi)$ around p, we have that $\mathcal{U}$ is homeomorphic to an open set $\varphi(\mathcal{U})$ contained in $\mathbb{R}^n$. Then, one could attempt to establish the differentiability of $f\colon \mathcal{M} \to \mathbb{R}$ at p in $\mathcal{U}$ by considering the differentiability of

$$f \circ \varphi^{-1}\colon \varphi(\mathcal{U}) \subset \mathbb{R}^n \to \mathbb{R}$$

at $\varphi(p)$. However, it is worth recognizing that more than one chart around p might be available, and the concept of differentiability for f at p should remain consistent irrespective of the chosen chart around p. This leads to the following definition.

Definition 2 (C^r (Differentiable) Manifold)**.** Let r be a natural number or ∞. A topological space $\mathcal{M}$ is a C^r *(differentiable) manifold* if it satisfies the following conditions: $\mathcal{M}$ is a topological manifold with an atlas $\{(\mathcal{U}_\lambda, \varphi_\lambda)\}_{\lambda\in\Lambda}$; and, for any $\alpha, \beta \in \Lambda$ where $\mathcal{U}_\alpha \cap \mathcal{U}_\beta \neq \emptyset$, the coordinate transformation

$$\varphi_\beta \circ \varphi_\alpha^{-1}\colon \varphi_\alpha\left(\mathcal{U}_\alpha \cap \mathcal{U}_\beta\right) \subset \mathbb{R}^n \to \varphi_\beta\left(\mathcal{U}_\alpha \cap \mathcal{U}_\beta\right) \subset \mathbb{R}^n \tag{2.2}$$

is of class C^r in the usual Euclidean sense. This atlas is then referred to as a C^r *atlas*.

In fact, if $f \circ \varphi_\beta^{-1}$ belongs to the class C^s for a certain integer $s \leq r$ at $\varphi_\beta(p)$, then $f \circ \varphi_\alpha^{-1} = (f \circ \varphi_\beta^{-1}) \circ (\varphi_\beta \circ \varphi_\alpha^{-1})$ is also of class C^s at $\varphi_\alpha(p)$, in accordance with the results established in the Euclidean space. Thus, the following definition is well-defined.

Definition 3 (Smoothness of $f\colon \mathcal{M} \to \mathbb{R}$)**.** Let $\mathcal{M}$ be a C^r manifold and s be an integer with $s \leq r$. A function $f\colon \mathcal{M} \to \mathbb{R}$ is said to be of class C^s on $\mathcal{M}$ if, for every point $p \in \mathcal{M}$, there exists a chart $(\mathcal{U}, \varphi)$ and $f \circ \varphi^{-1}$ is of class C^s at the point $\varphi(p)$.

Remark 1. Note that *differentiability* refers to whether a function is s times (continuously) differentiable at a point or over a set. Differentiability itself is not a s-th-order derivative. For example, the differentiability of f at p (i.e., the differentiability of $f \circ \varphi^{-1}$ at $\varphi(p)$) does not depend on the chart choice, but its (first-order) derivative

$$\frac{\partial \left(f \circ \varphi^{-1}\right)}{\partial x_i}(\varphi(p)), i = 1, \ldots, n \tag{2.3}$$

depend on the chart choice.

The Definition 2 arises from the discussion about the necessity for the differentiability of the function $f\colon \mathcal{M} \to \mathbb{R}$ to be consistently defined, irrespective of the selected chart. Actually, the smoothness of coordinate transformation $\varphi_\beta \circ \varphi_\alpha^{-1}$ in (2.2) is also important to define the differentiability of $F\colon \mathcal{M} \to \mathcal{N}$ between any two manifolds $\mathcal{M}$ and $\mathcal{N}$, as we will see later on (see Definition 5 later).

Example 1. Equipped with the atlas $\mathfrak{A} := \{(\mathbb{R}^n, \mathrm{id})\}$ consisting of a single chart, the Euclidean space $\mathbb{R}^n$ is an n-dimensional C^∞ manifold.

2.2.3 Smooth Structure

Let us explore the concept of atlases in more detail. A C^r manifold $\mathcal{M}$ can admit several C^r atlases. Suppose that $\mathcal{M}$ has two atlases: $\mathfrak{A} = \{(\mathcal{U}_\alpha, \varphi_\alpha)\}_{\alpha \in A}$ and $\mathfrak{B} = \{(\mathcal{V}_\beta, \psi_\beta)\}_{\beta \in B}$. If for any $\alpha \in A$ and $\beta \in B$ where $\mathcal{U}_\alpha \cap \mathcal{V}_\beta \neq \emptyset$, both $\psi_\beta \circ \varphi_\alpha^{-1}$ and $\varphi_\alpha \circ \psi_\beta^{-1}$ are of class C^r, we say that atlas $\mathfrak{B}$ is *equivalent* to atlas $\mathfrak{A}$. It is obvious that in this case $\mathfrak{A} \cup \mathfrak{B}$ is another valid C^r atlas for $\mathcal{M}$. On the other hand, if $f\colon \mathcal{M} \to \mathbb{R}$ is of C^s class for some $s \leq r$ (see Definition 3) with respect to atlas $\mathfrak{A}$, it also is of C^s with respect to atlas $\mathfrak{B}$. Hence, there is no need to specify one of them, and we consider the largest atlas that contains all of them.

Definition 4 (Maximal Atlas, Smooth Structure)**.** Consider a C^r manifold $\mathcal{M}$ equipped with an atlas $\mathfrak{A}$. The union of all atlases that are equivalent to $\mathfrak{A}$ is termed as the *maximal atlas* or *differential/smooth structure* of $\mathcal{M}$ generated by $\mathfrak{A}$. We assume the presence of a maximal atlas unless we endow a manifold with a specific atlas.

Remark 2. In general, manifold $\mathcal{M}$ may admit two distinct atlases that are not equivalent, so that their corresponding maximal atlases are also distinct. These atlases then lead to different smooth structures on $\mathcal{M}$. See example in [34, Exercise 8.14], where a function can qualify as smooth under one atlas but fail to be smooth under another atlas.

2.2.4 Differentiability

In a similar manner as Definition 3, we can define the differentiability for a general map between the manifolds $\mathcal{M}$ and $\mathcal{N}$.

Definition 5 (Smoothness of $F\colon \mathcal{M} \to \mathcal{N}$)**.** Consider $\mathcal{M}$ and $\mathcal{N}$ as m-dimensional and n-dimensional C^r manifolds, respectively. Let s be an integer with $s \leq r$. A function $F\colon \mathcal{M} \to \mathcal{N}$ is said to belong to class C^s on $\mathcal{M}$ if for any point $p \in \mathcal{M}$, there exist C^r charts $(\mathcal{U}, \varphi)$ and $(\mathcal{V}, \psi)$ for $\mathcal{M}$ and $\mathcal{N}$ containing points p and $f(p)$, respectively, such that $f(\mathcal{U}) \subset \mathcal{V}$, and the composition

$$\tilde{f} = \psi \circ f \circ \varphi^{-1}\colon \varphi(\mathcal{U}) \subset \mathbb{R}^m \to \psi(\mathcal{V}) \subset \mathbb{R}^n$$

is of class C^s at the point $\varphi(p)$ in the usual Euclidean sense.

As a special case of the above definition, we focus on curves on a manifold. Curves serve as essential tools for defining the concept of tangent vectors.

Definition 6 (Curve on Manifold)**.** Given an interval $I \subset \mathbb{R}$ and a C^r manifold $\mathcal{M}$, a map $\gamma\colon I \to \mathcal{M}$ is called a C^s curve, if it is of class C^s in the sense of Definition 5.

In the following, we adopt $r = s = \infty$, indicating that we primarily consider C^∞ manifolds, i.e., smooth manifolds and C^∞ (or smooth) functions and curves, unless we specify otherwise explicitly.

2.3 Tangent Vectors

At each point on a smooth manifold, we introduce a vector space known as a tangent space. This concept draws an analogy to the idea of a tangent plane at each point on a smooth surface.

Let us start by looking at a smooth curve $\gamma\colon I \to \mathbb{R}^n$ in Euclidean space, where I is an open interval of $\mathbb{R}$ containing 0. The curve goes through a point $p \in \mathbb{R}^n$ such that $\gamma(0) = p$. We can describe the curve as $\gamma(t) =: (x_1(t), x_2(t), \ldots, x_n(t))^T = \sum_{i=1}^n x_i(t)\mathbf{e}_i$, where $\mathbf{e}_i$ is the i-th standard basis vector in $\mathbb{R}^n$ and $x_i(0) = p_i$. Since $\gamma(t) - \gamma(0) \in \mathbb{R}^n$ for this curve, we can define the derivative of curve γ at 0 as

$$\gamma'(0) := \frac{\mathrm{d}}{\mathrm{d}t}\gamma(t)\bigg|_{t=0} = \lim_{t\to 0} \frac{\gamma(t) - \gamma(0)}{t} = \sum_{i=1}^n \frac{\mathrm{d}x_i}{\mathrm{d}t}(0)\mathbf{e}_i, \tag{2.4}$$

which is regarded as a tangent vector of the curve γ at $t = 0$. In physics, it is the velocity vector of a particle moving along the curve γ at time $t = 0$.

Next, let us consider a smooth curve $\gamma\colon I \to \mathcal{M}$ on an n-dimensional manifold $\mathcal{M}$. We want to extend the concept of a tangent vector to a curve on the manifold. For the points on manifold, however, the notion of linear structure, like $\gamma(t) - \gamma(0)$, is generally not well-defined. A natural idea would be this: given a chart $(\mathcal{U}, \varphi)$ around $\gamma(0) =: p \in \mathcal{M}$, one could represent the curve γ near p by $\varphi(\gamma(t)) =: (x_1(t), x_2(t), \ldots, x_n(t))^T =: x(t) \in \mathbb{R}^n$ and compute $(\varphi \circ \gamma)'(0) = x'(0)$ as in the Euclidean case (2.4). We then obtain

$$\frac{\mathrm{d}(\varphi(\gamma(t)))}{\mathrm{d}t}\bigg|_{t=0} = \begin{bmatrix} \frac{\mathrm{d}x_1}{\mathrm{d}t}(0) \\ \vdots \\ \frac{\mathrm{d}x_n}{\mathrm{d}t}(0) \end{bmatrix} \in \mathbb{R}^n. \tag{2.5}$$

However, the tangent vector in (2.5) is dependent on a specific chart. We need to define tangent vectors in a way that is independent of the choice of chart. Therefore, we formally define the tangent

vector as a differential operator of a real-valued function on the manfiold. Let $\mathfrak{F}_p(\mathcal{M})$ denote the set of all smooth real-valued functions defined on a neighborhood of $p \in \mathcal{M}$.

Definition 7 (Tangent Vectors)**.** Given an n-dimensional manifold $\mathcal{M}$, a map $\xi : \mathfrak{F}_p(\mathcal{M}) \to \mathbb{R}$ is called as a *tangent vector* at a point p on $\mathcal{M}$, if there exists a smooth curve $\gamma\colon I \to \mathcal{M}$ such that $\gamma(0) = p$ and the following holds:

$$\xi f = \frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\Big|_{t=0} \quad \text{for any } f \in \mathfrak{F}_p(\mathcal{M}). \tag{2.6}$$

Such a curve γ is said to *realize* the tangent vector ξ, we also write $\xi \equiv \dot{\gamma}(0)$.

We explain Definition 7 in more details. Let f be a smooth real-valued function defined on a neighborhood of $p \in \mathcal{M}$ and $\gamma\colon I \to \mathcal{M}$ be a smooth curve such that $\gamma(0) = p$. We examine the composition $f \circ \gamma\colon I \to \mathbb{R}$. Importantly, the derivative of $f \circ \gamma$ at $t = 0$ is a usual derivative that is irrelevant to the chart. However, we can expand the formulation of derivative utilizing a specific chart φ as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\Big|_{t=0} = \frac{\mathrm{d}}{\mathrm{d}t} (f \circ \varphi^{-1})(\varphi(\gamma(t)))\Big|_{t=0} = \sum_{i=1}^{n} \left\{ \frac{\mathrm{d}x_i}{\mathrm{d}t}(0)\, \frac{\partial \left(f \circ \varphi^{-1}\right)}{\partial x_i}(\varphi(p)) \right\}. \tag{2.7}$$

Note that $\frac{\mathrm{d}x_i}{\mathrm{d}t}(0)$ above is what we discussed in (2.5) and $\frac{\partial (f\circ\varphi^{-1})}{\partial x_i}(\varphi(p))$ is also discussed in (2.3). Both of them rely on the charts φ. Let $\frac{\partial}{\partial x_i}\Big|_p f := \frac{\partial (f\circ\varphi^{-1})}{\partial x_i}(\varphi(p))$. The we have

$$\frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\Big|_{t=0} = \sum_{i=1}^{n} \frac{\mathrm{d}x_i}{\mathrm{d}t}(0)\, \frac{\partial}{\partial x_i}\Big|_p f, \tag{2.8}$$

where we can interpret $\frac{\partial}{\partial x_i}\Big|_p$ as a map from $\mathfrak{F}_p(\mathcal{M})$ to $\mathbb{R}$. In conclusion, using different charts yields different expansions as (2.8). However, the result of the summation in right hand of (2.8) is guaranteed to be the same since the left hand of (2.8) is irrelevant to the charts. As the function f is chosen arbitrarily, the quantity $\frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\big|_{t=0}$ captures the nature of a tangent vector along γ.

Remark 3. Given any smooth curve γ on $\mathcal{M}$ with $\gamma(0) = p$, naturally, it yields a tangent vector, because we could simply define it according to (2.6). So it is often straightforward to write $\dot{\gamma}(0)$ to represent the tangent vectors.

Proposition 1 (Tangent Space)**.** *It can be shown that the set of all tangent vectors at point p can form an n-dimensional real vector space, referred to as the* tangent space *of $\mathcal{M}$ at p, denoted by $T_p\mathcal{M}$. To be specific, $T_p\mathcal{M}$ admits a linear structure defined by $(a\dot{\gamma}_1(0) + b\dot{\gamma}_2(0))\, f := a\,(\dot{\gamma}_1(0) f) + b\,(\dot{\gamma}_2(0) f)$ for any $\dot{\gamma}_1(0), \dot{\gamma}_2(0) \in T_p\mathcal{M}$ and $a, b \in \mathbb{R}$. See [5, Page 34-35] for details.*

Given a tangent vector ξ to $p \in \mathcal{M}$, there are infinitely many curves γ that can realize ξ. They can be characterized as follows [5, Proposition 3.5.2]:

$$\dot{\gamma}_1(0) = \dot{\gamma}_2(0), \text{ if and only if, } \frac{\mathrm{d}\left(\varphi\left(\gamma_1(t)\right)\right)}{\mathrm{d}t}\Big|_{t=0} = \frac{\mathrm{d}\left(\varphi\left(\gamma_2(t)\right)\right)}{\mathrm{d}t}\Big|_{t=0}$$

for some chart φ around p. Indeed, Boumal [34] defines tangent vectors as the the equivalence classes of curves using above property. That is, for any $c_1, c_2 \in C_p := \{c : c\colon I \to \mathcal{M}$ is smooth curves and

$c(0) = p\}$, we define $c_1 \sim c_2$ if $(\varphi \circ c_1)'(0) = (\varphi \circ c_2)'(0)$. Also, it can be shown that this equivalence relation is independent of the chart φ. Then, Boumal [34] defines tangent space as $T_p\mathcal{M} := C_p/\sim = \{[c] : c \in C_p\}$.

Example 2 (Canonical Identification)**.** Let us consider the case where $\mathcal{M} = \mathbb{R}^n$ and recall the definitions of $\gamma'(0)$ in (2.4). Utilizing Definition 7 and the chain rule in Euclidean case, we deduce that

$$\xi f = \dot{\gamma}(0)f = \frac{\mathrm{d}}{\mathrm{d}t} f(\gamma(t))\bigg|_{t=0} = \mathrm{D}f(p)\left[\gamma'(0)\right].$$

Notably, the map $x \mapsto \xi$, where $x = \gamma'(0)$ and $\xi = \dot{\gamma}(0)$ for some curve γ, stands as an isomorphism (a linear bijection) between $\mathbb{R}^n$ and $T_{\gamma(0)}\mathbb{R}^n$, and this relationship holds regardless of the chosen γ. It is useful to identify $\gamma'(0) \in \mathbb{R}^n$ with $\dot{\gamma}(0) \in T_{\gamma(0)}\mathbb{R}^n$. Consequently, we establish the canonical identification:

$$T_x\mathbb{R}^n \cong \mathbb{R}^n, \text{ for all } x \in \mathbb{R}^n. \tag{2.9}$$

We shall employ this identification when it becomes necessary. This result can be further extended to the general vector space, i.e.,

$$T_x\mathcal{E} \cong \mathcal{E}, \text{ for all } x \in \mathbb{R}^n. \tag{2.10}$$

Remark 4. Except for trivial cases such as Euclidean space, the tangent spaces of different points are usually different from each other. When $\mathcal{M}$ is a subset of Euclidean space, the tangent space is just a subspace.

2.4 Derivatives of Maps

In our previous sections, we have discussed the concept of differentiability as it applies to maps between manifolds. Building upon this understanding, in this section we shall introduce the formal definition of derivatives (see Remark 1 for difference between "differentiability" and "derivatives"). The derivatives can have a direct application in optimization algorithms on manifold.

Definition 8 (Derivatives, Differential)**.** Consider two smooth manifolds $\mathcal{M}$ and $\mathcal{N}$. The *derivative* or *differential* of a smooth map $F\colon \mathcal{M} \to \mathcal{N}$ at a point $p \in \mathcal{M}$ is defined by

$$\mathrm{D}F(p)\colon T_p\mathcal{M} \to T_{F(p)}\mathcal{N}, \ \mathrm{D}F(p)[\dot{\gamma}(0)] := \dot{\gamma}_F(0). \tag{2.11}$$

where γ is a curve on $\mathcal{M}$ with $\gamma(0) = p$, and $\gamma_F := F \circ \gamma$ is the corresponding image curve on $\mathcal{N}$.

It can be shown that $\mathrm{D}F(p)$ is linear. Let $\mathfrak{F}_{F(p)}(\mathcal{N})$ denote the set of all smooth real-valued functions defined on a neighborhood of $F(p) \in \mathcal{N}$. Note that for any $g \in \mathfrak{F}_{F(p)}(\mathcal{N})$, $g \circ F \in \mathfrak{F}_p(M)$. Let us take our attention to the behavior of the right-hand side of (2.11), $\dot{\gamma}_F(0)$, when it acts on $g \in \mathfrak{F}_{F(p)}(\mathcal{N})$. Then we have

$$\dot{\gamma}_F(0)(g) = \frac{\mathrm{d}(g \circ (F \circ \gamma))}{\mathrm{d}t}\bigg|_{t=0} = \frac{\mathrm{d}((g \circ F) \circ \gamma)}{\mathrm{d}t}\bigg|_{t=0} = \dot{\gamma}(0)(g \circ F).$$

This series of equalities show that the right-hand side of (2.11) is invariant with respect to the choice of curve γ provided they realize the same tangent vector. Without using the curve explicitly to realize

a tangent vector, we can also define the derivative as follows: for any $\xi \in T_p\mathcal{M}$, $\mathrm{D}F(p)[\xi]$ is a tangent vector in $T_{F(p)}\mathcal{N}$ such that for all g $\in \mathfrak{F}_{F(p)}(\mathcal{N})$,

$$(\mathrm{D}F(p)[\xi])(g) = \xi(g \circ F). \tag{2.12}$$

In fact, Definition 8 is consistent with the usual derivatives. If $F\colon \mathcal{E} \to \mathcal{E}'$ is a smooth function between two vector spaces, with identification $T_p\mathcal{E} \cong \mathcal{E}$ and $T_{F(p)}\mathcal{E}' \cong \mathcal{E}'$ from (2.10), then the differential $\mathrm{D}F(p)$ reduces to its classical directional derivative given by

$$\mathrm{D}F(p)\colon \mathcal{E} \to \mathcal{E}',\ \mathrm{D}F(p)\,[\xi] = \lim_{t\to 0} \frac{F(p+t\xi) - F(p)}{t}. \tag{2.13}$$

On the other hand, consider a smooth function $f\colon \mathcal{M} \to \mathbb{R}$. By identifying $T_{f(p)}\mathbb{R} \cong \mathbb{R}$, we have

$$\mathrm{D}f(p)\colon T_p\mathcal{M} \to \mathbb{R},\ \mathrm{D}f(p)[\dot\gamma(0)] = (f \circ \gamma)'(0). \tag{2.14}$$

We conclude this section with the following useful properties of derivatives, which are the natural extension of the classical results.

1. (Linearity of Smooth Maps) Consider smooth maps $F_1, F_2 : \mathcal{M} \to \mathcal{E}$, where $\mathcal{E}$ is a linear space, and real numbers a_1, a_2. For map $F\colon x \mapsto a_1F_1(x) + a_2F_2(x)$, we have

$$\mathrm{D}F(x) = a_1\mathrm{D}F_1(x) + a_2\mathrm{D}F_2(x).$$

2. (Chain Rule for Differentials [165, Proposition 3.2]) Let $F\colon \mathcal{M} \to \mathcal{M}'$ and $G : \mathcal{M}' \to \mathcal{M}''$ be smooth maps between manifolds. Then, the chain rule is applicable to the differential of their composition:

$$\mathrm{D}(G \circ F)(x)[v] = \mathrm{D}G(F(x))[\mathrm{D}F(x)[v]].$$

2.5 Embedded Submanifold

This section introduces the concept of a submanifold, which is a manifold in its own right, and at the same time a subset of another manifold (usually a subset of some vector spaces such as $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$).

Definition 9 ((Regular) Submanifold)**.** Let $\mathcal{M}$ be an m-dimensional manifold and $\mathcal{N}$ be a subset of $\mathcal{M}$. The subset $\mathcal{N} \subset \mathcal{M}$ is called an *n-dimensional (regular) submanifold of $\mathcal{M}$* if, for all $p \in \mathcal{N}$, there exists a chart $(\mathcal{U}; x_1, x_2, \ldots, x_m)$ of $\mathcal{N}$ containing p such that the intersection $\mathcal{N} \cap \mathcal{U} = \{q \in \mathcal{U} : x_{n+1}(q) = x_{n+2}(q) = \cdots = x_m(q) = 0\}$.

According to [193, Proposition 9.4], a regular submanifold $\mathcal{N}$ of $\mathcal{M}$ is indeed a manifold. In general, it is difficult to check whether a subset of a manifold is a regular submanifold through definition above. It will be convenient if we can show that some specific subset of a manifold is a submanifold without using local chart $(\mathcal{U}; x_1, x_2, \ldots, x_m)$. The following Theorem 1 and Theorem 2 state two useful and convenient conclusions. We first introduce a map called embedding as follows.

Definition 10 (Embedding)**.** Let $\mathcal{M}$ and $\mathcal{N}$ be smooth manifolds. A smooth map $F\colon \mathcal{M} \to \mathcal{N}$ is called an *embedding* if

(i) it is an immersion, namely, $\mathrm{D}F(p)\colon T_p\mathcal{M} \to T_{F(p)}\mathcal{N}$ is injective (i.e., rank of $\mathrm{D}F(p)$ equals to $\dim \mathcal{M}$) for all $p \in \mathcal{M}$;

(ii) the image $F(\mathcal{M})$ with the subspace topology (induced by $\mathcal{N}$) is homeomorphic to $\mathcal{M}$ under F.

By the next theorem, a regular submanifold is also called an *embedded submanifold* because it must be the image of some embedding map; actually, they are one and the same thing. See [193, Theorem 11.13 & 11.14] for proofs. In this thesis, we refer to them simply as "submanifolds".

Theorem 1 (Embedded Submanifold = Regular Submanifold)**.** *Let $\mathcal{M}$ and $\mathcal{N}$ be smooth manifolds. If $F\colon \mathcal{N} \to \mathcal{M}$ is an embedding, then its image $F(\mathcal{N})$ is a regular submanifold of $\mathcal{M}$. If $\mathcal{N}$ is a regular submanifold of $\mathcal{M}$, then the inclusion map $\iota\colon \mathcal{N} \to \mathcal{M}$, $\iota(p) = p$, is an embedding. (Note that the image of ι is $\mathcal{N}$.)*

The next theorem introduces an important class of submanifolds that cover most of the manifolds we will encounter in applications. In fact, most manifolds in Table 3.1 can be proved to be manifolds by the next theorem. See [193, Theorem 9.9] for proof.

Theorem 2 (Regular Level Set Theorem)**.** *Let $F\colon \overline{\mathcal{M}} \to \mathcal{N}$ be a smooth map between manifolds, with $\dim \mathcal{M} = m$ and $\dim \mathcal{N} = n$ $(m \geq n)$. If a level set $\mathcal{M} := F^{-1}(\{q\})$ with $q \in \mathcal{N}$ is nonempty, and $\mathrm{D}F(p)\colon T_p\overline{\mathcal{M}} \to T_{F(p)}\mathcal{N}$ is surjective (i.e., rank of $\mathrm{D}F(p)$ equals to n) for all $p \in \mathcal{M}$, then $\mathcal{M}$ is an $(m-n)$-dimensional submanifold of $\overline{\mathcal{M}}$.*

Let $\mathcal{M}$ be a submanifold of a manifold $\overline{\mathcal{M}}$ (by definition, $\mathcal{M} \subset \overline{\mathcal{M}}$). Then tangent space $T_x\mathcal{M}$ at a point x on $\mathcal{M}$ can be expressed as a subspace of $T_x\overline{\mathcal{M}}$. Indeed, the inclusion map $\iota\colon \mathcal{M} \to \overline{\mathcal{M}}$ satisfies $\mathrm{D}\iota(x)[\xi] \in T_x\overline{\mathcal{M}}$ for $\xi \in T_x\mathcal{M}$. For any smooth function $\bar{f} \in \mathfrak{F}_x(\overline{\mathcal{M}})$ and its restriction $f := \bar{f}\big|_{\mathcal{M}} \in \mathfrak{F}_x(\mathcal{M})$, by (2.12), we have

$$(\mathrm{D}\iota(x)[\xi])\bar{f} = \xi(\bar{f} \circ \iota) = \xi f.$$

Thus, we can identify ξ with $\mathrm{D}\iota(x)[\xi]$ and, thereby, $T_x\mathcal{M}$ with a subspace of $T_x\overline{\mathcal{M}}$. In fact, by [34, Corollary 8.76], we have

$$T_x\mathcal{M} = \ker \mathrm{D}F(x) \subset T_x\overline{\mathcal{M}}. \tag{2.15}$$

Specifically, we can identify $T_x\mathcal{M}$ with a subspace of $T_x\mathbb{R}^n \simeq \mathbb{R}^n$ if $\mathcal{M}$ is a submanifold of $\mathbb{R}^n$, and $T_X\mathcal{M}$ with a subspace of $T_X\mathbb{R}^{m\times n} \simeq \mathbb{R}^{m\times n}$ if $\mathcal{M}$ is a submanifold of $\mathbb{R}^{m\times n}$.

Example 3 (Stiefel Manifold)**.** For integers $k \leq n$, the set $\mathrm{St}(n,k) := \{X \in \mathbb{R}^{n\times k} : X^TX = I_k\}$ is called *(compact) Stiefel manifold*, where the columns of X are orthonormal in $\mathbb{R}^n$ with respect to the standard inner product $\langle x, y\rangle = x^Ty$. Notice that $\mathrm{St}(n,k) = F^{-1}(\{I_k\})$ with

$$F\colon \mathbb{R}^{n\times k} \to \mathrm{S}(k)\colon X \mapsto F(X) := X^TX$$

where $\mathrm{S}(k)$ is the linear space of symmetric matrices of size k. From (2.13), we can compute

$$\mathrm{D}F(X)\colon \mathbb{R}^{n\times k} \to \mathrm{S}(k)\colon \xi \mapsto \mathrm{D}F(X)[\xi] = X^T\xi + \xi^TX.$$

To demonstrate $\mathrm{D}F(X)$ is surjective for any fixed $X \in \mathrm{St}(n,k)$, we find that for any $A \in \mathrm{S}(k)$, there exists $\xi := \frac{1}{2}XA \in \mathbb{R}^{n\times k}$ such that $\mathrm{D}F(X)[\xi] = A$. Thus, by Theorem 2, $\mathrm{St}(n,k)$ is a submanifold of

$\mathbb{R}^{n\times k}$ and $\dim \mathrm{St}(n,k) = \dim \mathbb{R}^{n\times k} - \dim \mathrm{S}(k) = nk - \frac{k(k+1)}{2}$. Indeed, by (2.15), the tangent space $T_X \mathrm{St}(n,k) = \ker \mathrm{D}F(X)$ is given by

$$\begin{aligned} T_X \mathrm{St}(n,k) &= \left\{\xi \in \mathbb{R}^{n\times k} : X^T\xi + \xi^T X = 0\right\} \\ &= \left\{X\Omega + X_\perp B : \Omega \in \mathrm{Skew}(k), B \in \mathbb{R}^{(n-k)\times k}\right\}, \end{aligned}$$

where $\mathrm{Skew}(k) := \{\Omega \in \mathbb{R}^{k\times k} : \Omega^T = -\Omega\}$ is the linear space of skew-symmetric matrices of size k; $X_\perp \in \mathbb{R}^{n\times(n-k)}$ is an arbitrary matrix such that $[X, X_\perp] \in \mathbb{R}^{n\times n}$ is orthogonal. The second line above allows an explicit form by parametrization. Refer to [34, Section 7.3] and [5, Example 3.5.2] for detailed instructions. In particular, we consider two special cases of Stiefel manifold below.

- If $k = 1$, then $\mathrm{Sp}^{n-1} := \mathrm{St}(n,1) = \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$ is called *(unit) sphere manifold*. We have $\dim \mathrm{Sp}^{n-1} = n-1$ and
$$T_x \mathrm{Sp}^{n-1} = \left\{\xi \in \mathbb{R}^n : x^T\xi = 0\right\}.$$
- If $k = n$, then $\mathrm{O}(n) := \mathrm{St}(n,n) = \left\{X \in \mathbb{R}^{n\times n} : X^TX = I_n\right\}$ is called *orthogonal group*. It is a *group* equipped with matrix multiplication as its group operation. Being both a manifold and a group, it is called a *Lie group*. We have $\dim \mathrm{O}(n) = \frac{n(n-1)}{2}$ and
$$T_X\mathrm{O}(n) = \left\{X\Omega \in \mathbb{R}^{n\times n} : \Omega \in \mathrm{Skew}(n)\right\} = X\,\mathrm{Skew}(n).$$

2.6 Vector Fields

The concept of tangent bundle forms the basis for defining vector fields, and other fundamental concepts (e.g., retractions) in the study of Riemannian optimization.

Definition 11 (Tangent Bundle)**.** The *tangent bundle* $T\mathcal{M}$ of an n-dimensional manifold $\mathcal{M}$ is the set defined as

$$T\mathcal{M} = \{(x,v) : x \in \mathcal{M} \text{ and } v \in T_x\mathcal{M}\},$$

i.e., the disjoint union of all the tangent spaces.

Remark 5. When the context is clear, we may simplify the notation, using $v \in T\mathcal{M}$ to represent the pair $(x,v) \in T\mathcal{M}$.

A natural projection map $\pi : T\mathcal{M} \to \mathcal{M}, \pi(x,v) := x$ extracts the base point associated with a given tangent vector. It can be shown that the tangent bundle $T\mathcal{M}$ is itself a manifold, possessing a dimension of $2n$, in such a way that the projection map $\pi : T\mathcal{M} \to \mathcal{M}$ is smooth, where the smoothness is defined through Definition 5. See [127, Proposition 3.18] for details. Now, the manifold structure imposed on $T\mathcal{M}$ enables the definition of smooth vector fields on manifolds.

Definition 12 (Vector Field)**.** A *vector field* on the manifold $\mathcal{M}$ is a map, denoted as

$$V\colon \mathcal{M} \to T\mathcal{M},$$

such that $\pi \circ V = \mathrm{Id}_{\mathcal{M}}$, where $\mathrm{Id}_{\mathcal{M}}$ is the identity map on $\mathcal{M}$. This means that $V(x)$ is an element of the tangent space $T_x\mathcal{M}$ for all $x \in \mathcal{M}$. If such map V is smooth, then V is called a smooth vector field. The set of all smooth vector fields on $\mathcal{M}$ is denoted by $\mathfrak{X}(\mathcal{M})$.

Example 4. Consider $\mathbb{R}^n$ with canonical identification $T_x\mathbb{R}^n \cong \mathbb{R}^n$ for all $x \in \mathbb{R}^n$. Then $V \in \mathfrak{X}(\mathbb{R}^n)$ is exactly a smooth function $V : \mathbb{R}^n \to \mathbb{R}^n$.

Remark 6. A scalar field is described as a function that assigns a scalar value to every point in a specific space, such as Euclidean space or a manifold. In the context of Riemannian optimization, it precisely corresponds to the objective function of our problem $\min_{x\in\mathcal{M}} f(x)$. We often conflate terminologies for "real-valued function" and "scalar field".

Smooth vector fields play an important role in the Riemannian optimization. We summarize the following characterization of smooth vector fields. Let $\mathfrak{F}(\mathcal{M})$ denote the set of all smooth real-valued function defined on the manifold $\mathcal{M}$.

Definition 13 (Action of Vector Field on Scalar Field)**.** Let V be a vector field on the manifold $\mathcal{M}$. The action of V on a smooth scalar field $f \in \mathfrak{F}(\mathcal{M})$ is defined as

$$(Vf)(x) := \mathrm{D}f(x)[V(x)], \tag{2.16}$$

where the right-hand above is given by (2.14).

An important result is that a vector field V on $\mathcal{M}$ is smooth if and only if $Vf \colon \mathcal{M} \to \mathbb{R}$ is smooth for all $f \in \mathfrak{F}(\mathcal{M})$. See [127, Proposition 8.14]. This allows us to regard $V \in \mathfrak{X}(\mathcal{M})$ as a map

$$V \colon \mathfrak{F}(\mathcal{M}) \to \mathfrak{F}(\mathcal{M}).$$

Definition 14 (Multiplication of Scalar Field and Vector Field)**.** For any $f \in \mathfrak{F}(\mathcal{M})$ and $V \in \mathfrak{X}(\mathcal{M})$, the vector field fV on $\mathcal{M}$, defined as

$$(fV)(x) := f(x)V(x). \tag{2.17}$$

By [34, Exercise 3.46 & 8.48], fV is also smooth. This allows us to regard $f \in \mathfrak{F}(\mathcal{M})$ as a map $f \colon \mathfrak{X}(\mathcal{M}) \to \mathfrak{X}(\mathcal{M})$, performing a pointwise scaling. Be cautious not to conflate the notations of Vf in (2.16) and fV in (2.17).

Remark 7. It can be shown that for any $V, W \in \mathfrak{X}(\mathcal{M})$ and $f, g \in \mathfrak{F}(\mathcal{M})$, the vector field $fV + gW$ is also smooth. Thus, $\mathfrak{X}(\mathcal{M})$ forms a module over $\mathfrak{F}(\mathcal{M})$. When f, g is taken to be constant, $\mathfrak{X}(\mathcal{M})$ is also a real vector space.

2.7 Retractions

A retraction is a key operation that provides a way to move around the manifold, and it plays a central role in Riemannian optimization algorithms.

Definition 15 (Retraction - I)**.** A *retraction* on a manifold $\mathcal{M}$ is a smooth map

$$\mathrm{R}\colon T\mathcal{M} \to \mathcal{M} : (x, \xi) \mapsto \mathrm{R}_x(\xi)$$

such that for each $(x, \xi) \in T\mathcal{M}$, the curve $\gamma(t) := \mathrm{R}_x(t\xi)$ satisfies $\dot{\gamma}(0) = \xi$.

Frequently, $\mathrm{R}_x\colon T_x\mathcal{M} \to \mathcal{M}$ denotes the restriction of R to $T_x\mathcal{M}$. Let $(x, \xi) \in T\mathcal{M}$, i.e., $x \in \mathcal{M}$ and $\xi \in T_x\mathcal{M}$. If the curve $\gamma(t) = \mathrm{R}_x(t\xi)$ realize the tangent vector ξ, it is implicitly required that $\gamma(0) = \mathrm{R}_x(0_x) = x$. This means that the retraction R_x maps the zero element 0_x of the tangent space $T_x\mathcal{M}$ to the point x itself. On the other hand, using chain rule and the canonical identification $T_{0_x}T_x\mathcal{M} \cong T_x\mathcal{M}$, we have $\dot{\gamma}(0) = \mathrm{D}\mathrm{R}_x(0_x)[\xi] = \xi$. Note that ξ is arbitrary. Thus, an equivalent definition of retraction is given as follows.

Definition 16 (Retraction - II)**.** A retraction on a manifold $\mathcal{M}$ is a smooth map $\mathrm{R}\colon T\mathcal{M} \to \mathcal{M}$ with the following properties, where R_x denotes the restriction of R to $T_x\mathcal{M}$:

1. $\mathrm{R}_x(0_x) = x$.
2. $\mathrm{D}\mathrm{R}_x(0_x) = \mathrm{Id}_{T_x\mathcal{M}}$.

The second condition implies that the differential of R_x at 0_x is the identity map $\mathrm{Id}_{T_x\mathcal{M}}$ on $T_x\mathcal{M}$. It means that R_x behaves like the identity map on the tangent space $T_x\mathcal{M}$ near the origin 0_x.

Example 5. When $\mathcal{M} = \mathbb{R}^n$, $\mathrm{R}_x(\xi) = x + \xi$ is a trivial retraction.

Example 6 (Retraction on Sphere)**.** The simplest retraction on sphere manifold Sp^{n-1} is

$$\mathrm{R}_x(\xi) = \frac{x + \xi}{\|x + \xi\|}.$$

It is straightforward to check that this retraction satisfies the properties in Definition 16.

Example 7 (Retractions on Stiefel Manifold)**.** There are many often-used retractions on Stiefel manifold $\mathrm{St}(n, k)$. Let $X \in \mathrm{St}(n, k)$ and $\xi \in T_X\,\mathrm{St}(n, k)$ (Recall Example 3).

- Q-factor retraction:

  $$\mathrm{R}_X(\xi) = Q,$$

  where $QR = X + \xi$ is a thin/reduced QR decomposition with $Q \in \mathrm{St}(n, k)$ and $R \in \mathbb{R}^{k\times k}$ upper triangular with nonnegative diagonal. Since

  $$(X + \xi)^T(X + \xi) = I_k + \xi^T\xi \succ 0 \tag{2.18}$$

  for all $\xi \in T_X\,\mathrm{St}(n, k)$, then $X + \xi$ has full rank k, implying that the thin/reduced QR decomposition is uniquely determined [96, Theorem 2.1.14]. To see this retraction satisfies the properties in Definition 16, refer to [5, Example 8.1.5].
- Polar retraction:

  $$\mathrm{R}_X(\xi) = (X + \xi)\left((X + \xi)^T(X + \xi)\right)^{-1/2} = (X + \xi)\left(I_k + \xi^T\xi\right)^{-1/2} \tag{2.19}$$

  $$= U, \tag{2.20}$$

where $M^{-1/2}$ denotes the inverse of square root of M; and $US = X + \xi$ is a polar decomposition with $U \in \mathrm{St}(n,k)$ and positive semidefinite matrix $S \in \mathrm{S}(k)$. Again by (2.18), $X + \xi$ has full rank k, implying that the matrix U (called polar factor) of polar decomposition is uniquely determined [96, Theorem 7.3.1, (b)-(c)]. It is straightforward to check that this retraction (2.19) satisfies the properties in Definition 16.

- Cayley transform retraction [201, 219, 115]:

$$\mathrm{R}_X(\xi) = \left(I - \frac{1}{2}W_\xi\right)^{-1}\left(I + \frac{1}{2}W_\xi\right)X$$

where $\mathrm{Skew}(n) \ni W_\xi := P_X \xi X^T - X\xi^T P_X$ and $P_X := I - \frac{1}{2}XX^T$. It can be shown that the curve $t \mapsto \gamma(t) = \mathrm{R}_X(t\xi)$ is contained in $\mathrm{St}(n,k)$ and satisfies $\gamma(0) = X$ and $\dot\gamma(0) = W_\xi X = \xi$ as in Definition 15, see [201, Lemma 3].

2.8 Riemannian Manifolds

In order to define geometric structures on manifolds, such as the lengths of curves and the distances between points, we now introduce the notion of Riemannian metrics.

Since tangent spaces are vector spaces, we can endow them with the inner products. Given a tangent space $T_x\mathcal{M}$ at a point x on a manifold $\mathcal{M}$, an *inner product* $\langle\cdot,\cdot\rangle_x : T_x\mathcal{M} \times T_x\mathcal{M} \to \mathbb{R}$ is a function that satisfies:

1. (Non-negativity) $\langle v, v\rangle_x \geq 0$ for all $v \in T_x\mathcal{M}$.
2. (Definiteness) $\langle v, v\rangle_x = 0$ if and only if $v = 0_x$.
3. (Symmetry) $\langle u, v\rangle_x = \langle v, u\rangle_x$ for all $u, v \in T_x\mathcal{M}$.
4. (Linearity) $\langle au + bv, w\rangle_x = a\langle u, w\rangle_x + b\langle v, w\rangle_x$ for all $u, v, w \in T_x\mathcal{M}$ and $a, b \in \mathbb{R}$.

With this inner product, we can induce a norm as $\|\xi\|_x := \sqrt{\langle \xi, \xi\rangle_x}$ on $T_x\mathcal{M}$. For convenience, we may omit the subscript x when it is clear from the context.

2.8.1 Riemannian Metric

Definition 17 (Riemannian Metric)**.** A *Riemannian metric* on the manifold $\mathcal{M}$ is a choice of inner product $\langle\cdot,\cdot\rangle_x$ for each point x on $\mathcal{M}$ such that $x \mapsto \langle\cdot,\cdot\rangle_x$ varies smoothly across the manifold in the following sense: for all $V, W \in \mathfrak{X}(\mathcal{M})$, the map

$$x \mapsto \langle V(x), W(x)\rangle_x \tag{2.21}$$

is a smooth function from $\mathcal{M}$ to $\mathbb{R}$. (See Fig. 2.1) When the manifold $\mathcal{M}$ is equipped with a Riemannian metric, $\mathcal{M}$ is called a *Riemannian manifold*.

Example 8. Consider $\mathcal{M} = \mathbb{R}^n$ with canonical identification $T_x\mathbb{R}^n \cong \mathbb{R}^n$, the standard Euclidean product $\langle v, w\rangle_x := v^T w$ (the same at all points) is a Riemannian metric.

It is well-known that for any smooth manifold $\mathcal{M}$, there exists a Riemannian metric on $\mathcal{M}$; and thus, any smooth manifold can become a Riemannian manifold. For more detail, see [127, Proposition 13.3].

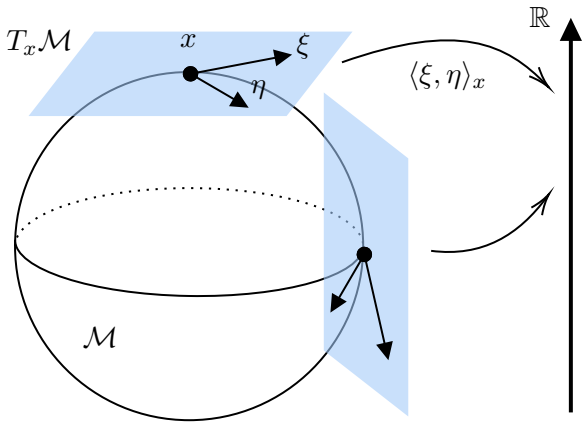


Fig. 2.1 An illustration of Riemannian metric on sphere manifold.

Recall that the second-countability is the topological property that we request for defining the manifold (review Definition 1). This property ensures the manifold has a countable base for its topology. This is a crucial requirement for various mathematical constructions and proofs, including the existence of *partitions of unity*, which are used for the construction of Riemannian metrics.

2.8.2 Local Frame

Next, we will introduce the local frame, which is a technical tool and very useful in many proofs.

Definition 18 (Local Frame)**.** A *local frame* around a point x on an n-dimensional manifold $\mathcal{M}$ is a collection of n smooth vector fields, $\{E_1, E_2, \ldots, E_n\}$, defined on a neighborhood $\mathcal{U}$ of x such that for every point $y \in \mathcal{U}$, the tangent vectors $\{E_1(y), E_2(y), \ldots, E_n(y)\}$ form a basis for $T_y\mathcal{M}$.

It can be shown that there exists a local frame around any $x \in \mathcal{M}$. Moreover, consider $\mathcal{M}$ as a Riemannian manifold. Use the Gram-Schmidt procedure on an existing local frame, an *orthonormal* local frame exists [127, Corollary 13.8]. Specifically, an orthonormal local frame $\{E_1, E_2, \ldots, E_n\}$ is defined on a neighborhood $\mathcal{U}$ around x such that

$$\forall y \in \mathcal{U}, \quad \langle E_i(y), E_j(y) \rangle_y = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases}$$

2.8.3 Riemannian Gradient

We next define the Riemannian gradient of a scalar field f as a vector field, denoted by $\operatorname{grad} f$.

Definition 19 (Riemannian Gradient)**.** The *Riemannian gradient* of f at x, $\operatorname{grad} f(x)$, is the unique tangent vector in $T_x\mathcal{M}$ characterized by the following identities:

$$\mathrm{D}f(x)[\xi] = \langle \xi, \operatorname{grad} f(x) \rangle, \tag{2.22}$$

where $\mathrm{D}f(x) \colon T_x\mathcal{M} \to \mathbb{R}$ represents the differential of the scalar field f at the point x.

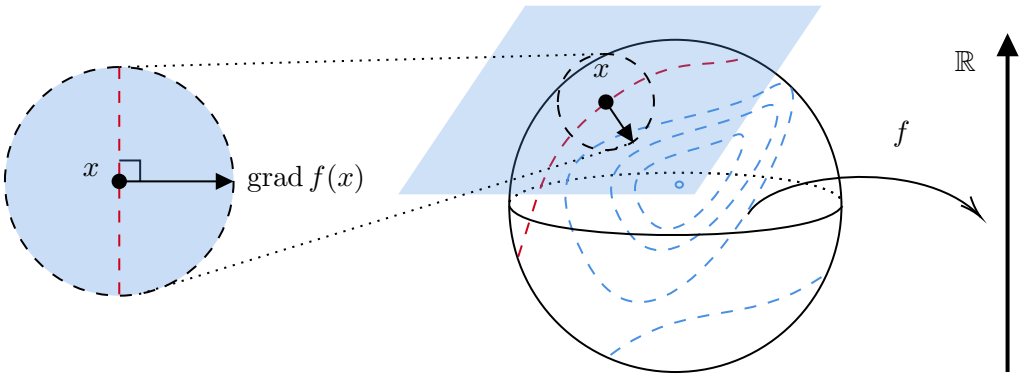


Fig. 2.2 Illustration of the Riemannian gradient on sphere manifold.

Note that the existence and uniqueness of $\operatorname{grad} f(x)$ comes from the *Riesz representation theorem* (see, e.g., [9, 6.42]). We can show that the direction (i.e., tangent vector) of $\operatorname{grad} f(x)$ is the steepest ascent direction of f at x in the sense of

$$\frac{\operatorname{grad} f(x)}{\|\operatorname{grad} f(x)\|} = \underset{\xi \in T_x\mathcal{M}:\|\xi\|=1}{\arg\max} \mathrm{D}f(x)[\xi]. \tag{2.23}$$

Recall the Cauchy–Schwarz inequality on general vector space: $|\langle u, v\rangle| \leq \|u\|\|v\|$ and equality holds if and only if $v = \lambda u$ for some constant λ. Then from the definition of (2.22), the objective of (2.23) is maximal when $\xi^* = \lambda \operatorname{grad} f(x)$ for some positive number λ. Taking norm in both sides leads $1 = \|\xi^*\| = \lambda\|\operatorname{grad} f(x)\|$, and hence $\xi^* = \operatorname{grad} f(x)/\|\operatorname{grad}\|$. Simimarirly, $-\operatorname{grad} f(x)$ is the steepest descent direction of f at x. For example, see Fig. 2.2.

Remarkably, for any $f \in \mathfrak{F}(\mathcal{M})$, the *gradient vector field* (Fig. 2.3 illustrates a simple example on 2-dim sphere Sp^2.)

$$x \mapsto \operatorname{grad} f(x)$$

is a smooth vector field on $\mathcal{M}$, see [34, Proposition 8.58]. Consequently, the following scalar field on $\mathcal{M}$:

$$x \mapsto \|\operatorname{grad} f(x)\|_x^2 = \langle \operatorname{grad} f(x), \operatorname{grad} f(x)\rangle_x \tag{2.24}$$

is also smooth, owing to the smoothness of the Riemannian metric as previously stated in (2.21). The smoothness of this type of function is particularly useful in analyzing convergence in later chapters.

2.8.4 Riemannian submanifold

In Section 2.5, we discussed the embedded submanifold, which is a manifold itself and also is a subset of some Euclidean space $\mathcal{E}$ (often called *ambient* space). Note that although the ambient space can be a general manifold (as in Section 2.5), henceforth it suffices for us to focus only on the Euclidean space $\mathcal{E}$ such as $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$. We next introduce the most important class of Riemannian manifolds, called Riemannian submanifold of a Euclidean space $\mathcal{E}$. Let $\mathcal{E}$ equipped with an inner product $\langle\cdot,\cdot\rangle$, and just like in Example 8, we call it the Euclidean metric.

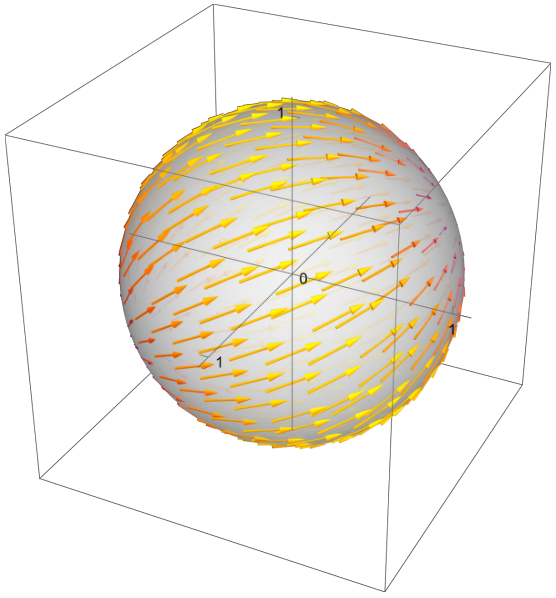


Fig. 2.3 Riemannian gradient field of $f(x) = -x_1 + 2x_2 + x_3$ on 2-dim sphere Sp^2.

Definition 20 (Riemannian Submanifold)**.** Let $\mathcal{M}$ be an embedded submanifold of $\mathcal{E}$. Equipped with the Riemannian metric obtained by restriction of the metric of $\mathcal{E}$, we call $\mathcal{M}$ a Riemannian submanifold of $\mathcal{E}$.

As we discussed in Section 2.5, we know from (2.15) that $T_x\mathcal{M}$ is a subspace of $T_x\mathcal{E} \cong \mathcal{E}$. Hence, we can let $T_x\mathcal{M}$ inherit the same inner product form ambient space $\mathcal{E}$. In this case, we can conjecture that the geometric tools of the Riemannian submanifold are directly related to their counterparts in Euclidean space. Indeed, from [34, Proposition 3.61], we have the following nice conclusion.

Proposition 2 (Gradient of Riemannian Submanifold)**.** *Let $\mathcal{M}$ be a Riemannian submanifold of $\mathcal{E}$ endowed with the metric $\langle\cdot,\cdot\rangle$ and let $f\colon \mathcal{M} \to \mathbb{R}$ be a smooth function. Let Proj_x denote the orthogonal projection form $\mathcal{E}$ onto subspace $T_x\mathcal{M}$ with respect to $\langle\cdot,\cdot\rangle$. Then Riemannian gradient of f is given by*

$$\operatorname{grad} f(x) = \mathrm{Proj}_x(\operatorname{grad} \tilde{f}(x)),$$

where $\tilde{f}$ is any smooth extension of f to a neighborhood of $\mathcal{M}$ in $\mathcal{E}$.

Remark 8. In most cases of applications, the smooth function f is also well-defined on the whole $\mathcal{E}$. Then, the smooth extension of $\tilde{f}$ is f itself. See [34, Example 3.62] for a concrete example on sphere manifold. In fact, we used the above proposition to compute and plot the gradients of $f(x) = -x_1 + 2x_2 + x_3$ on 2-dim sphere $\mathrm{Sp}^2 = \{x \in \mathbb{R}^3 : \|x\|_2 = 1\}$ in Fig. 2.3.

2.9 Connections, Covariant Derivatives and Hessians

2.9.1 Riemannian Connections

In the context of Euclidean space, a vector field $V \in \mathfrak{X}(\mathbb{R}^n)$ is essentially a function from $\mathbb{R}^n$ to $\mathbb{R}^n$. Then we can differentiate vector fields using the classical directional derivative, i.e., for any $x \in \mathbb{R}^n$ and

$u \in \mathbb{R}^n \cong T_x\mathbb{R}^n$,

$$\mathrm{D}V(x)[u] = \lim_{t\to 0} \frac{V(x+tu) - V(x)}{t}, \tag{2.25}$$

However, when working on the general manifolds, we need a more sophisticated notion of derivative of vector fields. This is where the concept of an (affine) connection, denoted by ∇, comes in.

Definition 21 ((Affine) Connection)**.** An *(affine) connection* on $\mathcal{M}$ is defined as an operator

$$\nabla : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \to \mathfrak{X}(\mathcal{M}),\ (U, V) \mapsto \nabla_U V,$$

where given any vector fields $U, V, W \in \mathfrak{X}(\mathcal{M})$, and scalar fields $f, g \in \mathfrak{F}(\mathcal{M})$ along with $a, b \in \mathbb{R}$, operator ∇ satisfies the following properties:

1. ($\mathfrak{F}(\mathcal{M})$-linearity) $\nabla_{(fU+gW)}V = f\nabla_U V + g\nabla_W V$;
2. ($\mathbb{R}$-linearity) $\nabla_U(aV + bW) = a\nabla_U V + b\nabla_U W$;
3. (Leibniz rule) $\nabla_U(fV) = (Uf)V + f\nabla_U V$.

Any manifold has infinitely many connections [5, Proposition 5.2.1]. However, once we add a Riemannian structure to $\mathcal{M}$, a certain kind of connection (called Riemannian connection) satisfies the additional properties that make it the most appropriate connection. But before that, we need to introduce some new operations involving vector fields. In the following, let $\mathcal{M}$ be a Riemannian manifold with its Riemannian metric $\langle\cdot,\cdot\rangle$.

Inner Product of Vector Fields: Given $U, V \in \mathfrak{X}(\mathcal{M})$, let $\langle U, V\rangle \in \mathfrak{F}(\mathcal{M})$ be defined as: for all $x \in \mathcal{M}$,

$$\langle U, V\rangle(x) := \langle U(x), V(x)\rangle_x. \tag{2.26}$$

This operation defines an inner product between two vector fields, which results in a scalar field. Using this notation and recall (2.16) and (2.22), we can see that for any $f \in \mathfrak{F}(\mathcal{M})$,

$$Vf = \langle V, \operatorname{grad} f\rangle.$$

Lie Bracket of Vector Fields: Given $U, V \in \mathfrak{X}(\mathcal{M})$, the Lie bracket is defined as (recall (2.16)):

$$[U, V] : \mathfrak{F}(\mathcal{M}) \to \mathfrak{F}(\mathcal{M}),\ f \mapsto [U, V]f := U(Vf) - V(Uf).$$

This operation quantifies the extent to which the vector fields U and V do not commute. Remarkably, there exists a unique smooth vector field W on $\mathcal{M}$ such that $[U, V]f = Wf$ holds for all $f \in \mathfrak{F}(\mathcal{M})$ (see [34, Proposition 8.61]). We identify $[U, V]$ with that vector field W. Consequently, Lie bracket can be seem as a map

$$[\cdot,\cdot] : \mathfrak{X}(\mathcal{M}) \times \mathfrak{X}(\mathcal{M}) \to \mathfrak{X}(\mathcal{M}).$$

Now we are ready to define the Riemannian connection. Its existence and uniqueness is stated in the fundamental theorem of Riemannian geometry. See [126, Theorem 5.10], [34, Theorem 8.63].

Theorem 3 (Fundamental Theorem of Riemannian Geometry)**.** *Given a Riemannian manifold $\mathcal{M}$, there exists a unique connection ∇ that satisfies two additional properties for all vector fields $U, V, W \in \mathfrak{X}(\mathcal{M})$:*

4. *(Symmetry)* $[U, V] = \nabla_U V - \nabla_V U$;
5. *(Compatibility with Metric)* $U\langle V, W\rangle = \langle \nabla_U V, W\rangle + \langle V, \nabla_U W\rangle$.

This specific connection is called the Levi-Civita *or* Riemannian connection.

Example 9. Consider $\mathbb{R}^n$ with the Euclidean metric. Let U, V be smooth vector fields from $\mathbb{R}^n$ to $\mathbb{R}^n$. Then the Riemannian connection of $\mathbb{R}^n$ is

$$(\nabla_U V)(x) = \lim_{t\to 0} \frac{V(x + tU(x)) - V(x)}{t}. \tag{2.27}$$

This ∇ is called the *canonical Euclidean connection*.

2.9.2 Covariant Derivatives

Of particular note that, for any (affine) connection ∇ and vector fields $U, V \in \mathfrak{X}(\mathcal{M})$, the vector field $\nabla_U V$ at a point x only depends on the value of U at that point (see [34, Proposition 8.64]). More precisely, if we fix arbitrary $V \in \mathfrak{X}(\mathcal{M})$ and $x \in \mathcal{M}$, then for any $U_1, U_2 \in \mathfrak{X}(\mathcal{M})$ with $U_1(x) = U_2(x)$, we have $(\nabla_{U_1} V)(x) = (\nabla_{U_2} V)(x)$. Obviously, the connection given in (2.27) justifies this property in the Euclidean space. Based on such property, given $V \in \mathfrak{X}(\mathcal{M})$ and $u \in T_x\mathcal{M}$, we can write

$$\nabla_u V := (\nabla_U V)(x) \in T_x\mathcal{M}$$

for arbitrary $U \in \mathfrak{X}(\mathcal{M})$ with $U(x) = u$. This gives rise to the following very important concept.

Definition 22 (Covariant Derivative of Vector Field)**.** Let $\mathcal{M}$ be a manifold with a connection ∇ and $V \in \mathfrak{X}(\mathcal{M})$. The *covariant derivative of vector field V at a point $x \in \mathcal{M}$* is the linear operator

$$\nabla V(x)\colon T_x\mathcal{M} \to T_x\mathcal{M},\ \nabla V(x)[u] := \nabla_u V. \tag{2.28}$$

Then $x \mapsto \nabla V(x)$ is a map that assigns to each point x a linear operator from and to the tangent space of that point.

Example 10. For $\mathbb{R}^n$ with canonical Euclidean connection (2.27), the covariant derivative of vector field $V \in \mathfrak{X}(\mathbb{R}^n)$ at $x \in \mathbb{R}^n$ is $\nabla V(x)[u] = \mathrm{D}V(x)[u]$ given in (2.25). On the other hand, it is well-known that $\mathrm{D}V(x)[u] = \mathbf{J}_V(x)u$ for all $u \in \mathbb{R}^n$ where $\mathbf{J}_V(x) \in \mathbb{R}^{n\times n}$ is the Jacobian of V at x. Thus, the Jacobian $\mathbf{J}_V(x)$ identifies the linear operator $\nabla V(x)$.

2.9.3 Riemannian Hessian

In particular, the Riemannian Hessian of a smooth scalar field f is obtained when $V = \operatorname{grad} f$ in (2.28).

Definition 23 (Riemannian Hessian)**.** Let $\mathcal{M}$ be a Riemannian manifold with its Riemannian connection ∇. For a smooth function $f\colon \mathcal{M} \to \mathbb{R}$, the Riemannian Hessian of f at a point $x \in \mathcal{M}$ is the linear operator

$$\operatorname{Hess} f(x)\colon T_x\mathcal{M} \to T_x\mathcal{M},\ \operatorname{Hess} f(x)[u] := \nabla_u \operatorname{grad} f.$$

Equivalently, $\operatorname{Hess} f$ maps $\mathfrak{X}(\mathcal{M})$ to $\mathfrak{X}(\mathcal{M})$ as $\operatorname{Hess} f[U] := \nabla_U \operatorname{grad} f$.

Note that the Riemannian Hessian is self-adjoint with respect to the Riemannian metric. Specifically, for any point $x \in \mathcal{M}$ and any tangent vectors $u, v \in T_x\mathcal{M}$, the following equality holds:

$$\langle \mathrm{Hess}\, f(x)[u], v\rangle_x = \langle u, \mathrm{Hess}\, f(x)[v]\rangle_x.$$

Remark 9. Unless otherwise stated, we always equip Riemannian manifolds with Riemannian connections. Note that symbol ∇ denotes the connection rather than the usual Euclidean gradient. Instead, we use $\mathrm{egrad}\, f(x)$ to express the Euclidean gradient if necessary.

As in Proposition 2, for the smooth function f defined on a Riemannian submanifold of a Euclidean space $\mathcal{E}$, we can obtain the Riemannian Hessian by using the operations in Euclidean sense and then orthogonally project the result to the tangent spaces. From [34, Corollary 5.16], we have the following conclusion. See [34, Example 5.17] for a concrete example.

Proposition 3 (Hessian of Riemannian Submanifold)**.** *Let $\mathcal{M}$ be a Riemannian submanifold of $\mathcal{E}$ endowed with the metric $\langle\cdot,\cdot\rangle$ and let $f\colon \mathcal{M} \to \mathbb{R}$ be a smooth function. Let $\bar{G}$ be a smooth extension of* $\mathrm{grad}\, f$*, that is, $\bar{G}$ is any smooth vector field defined on a neighborhood of $\mathcal{M}$ in $\mathcal{E}$ such that $\bar{G}(x) = \mathrm{grad}\, f(x)$ for all $x \in \mathcal{M}$. Then,*

$$\mathrm{Hess}\, f(x)[u] = \mathrm{Proj}_x(\mathrm{D}\bar{G}(x)[u]).$$

Remark 10. It is important to note that the above proposition does not mean: compute the Euclidean Hessian and then orthogonally project it to tangent space. Instead, after computing the Riemannian gradient according to the Proposition 2, we will use classical operations to obtain the differential of the Riemannian gradient field $x \mapsto \mathrm{grad}\, f(x)$, and finally project it orthogonally. In most applications, the map $x \mapsto \mathrm{grad}\, f(x)$ is also well-defined over the whole $\mathcal{E}$. Thus, the smooth extension $\bar{G}$ is $\mathrm{grad}\, f$ itself.

2.10 Induced Covariant Derivative and Geodesics

> *"A body remains at rest, or in motion at a constant speed in a straight line, unless acted upon by a force."* - Newton's First Law of Motion [142]

In physics, a straight line can be thought of as a trajectory with zero acceleration. A *geodesic* intuitively coincides with a "straight line" on the manifold $\mathcal{M}$. For example, a great circle on sphere – the circular intersection of sphere with a plane passing through the sphere's center point. This section presents a rigorous definition of a geodesic, which requires the notion of *induced covariant derivatives*. Using the induced covariant derivative, we can defined the acceleration of a curve on a manifold, then a geodesic is a curve with zero acceleration.

Suppose that $c\colon I \to \mathcal{M}$ is a smooth curve on the manifold $\mathcal{M}$, where I is an open interval. A map

$$Z\colon I \to T\mathcal{M} \tag{2.29}$$

is called a *vector field along the curve* c if, for any t in I, its image $Z(t)$ lies in the tangent space $T_{c(t)}\mathcal{M}$ at $c(t) \in \mathcal{M}$. Furthermore, Z is called a *smooth* vector field along c if it is a smooth map. The collection of all smooth vector fields along the curve c is denoted by $\mathfrak{X}(c)$. There are two important types of smooth vector fields along c as follows.

(1) For any $U \in \mathfrak{X}(\mathcal{M})$, the composition $U \circ c \in \mathfrak{X}(c)$.

(2) For every t in I, the *velocity* of c at t is defined as

$$\dot{c}(t) := \dot{c}_t(0), \tag{2.30}$$

where c_t is the shifted curve given as $\tau \mapsto c_t(\tau) := c(t+\tau)$. The domain of c_t (the open interval containing 0) can shrink sufficiently such that it is well-defined. The right hand of (2.30) is a tangent vector at the point $c_t(0) = c(t)$ (see Remark 3). The *velocity vector field* of c is defined as

$$t \mapsto \dot{c}(t). \tag{2.31}$$

Then we have $\dot{c} \in \mathfrak{X}(c)$.

Let $\mathfrak{F}(I)$ denote the set of all smooth real-valued functions defined on the interval I. Then $g \in \mathfrak{F}(I)$ implies $g' \in \mathfrak{F}(I)$. Note that the set $\mathfrak{X}(c)$ forms a module over $\mathfrak{F}(I)$ under pointwise vector addition and pointwise multiplication, i.e., for any $Y, Z \in \mathfrak{X}(c)$ and $f, g \in \mathfrak{F}(I)$, we have $fY + gZ \in \mathfrak{X}(c)$. As we did in (2.26), given any $Y, Z \in \mathfrak{X}(c)$, we define $\langle Y, Z\rangle \in \mathfrak{F}(I)$ by

$$\langle Y, Z\rangle(t) := \langle Y(t), Z(t)\rangle_{c(t)} \text{ for all } t \in I. \tag{2.32}$$

Based on the above notations, we are ready to present the following theorem. See [34, Theorem 8.67] or [126, Theorem 4.24] for proofs.

Theorem 4 (Induced Covariant Derivative)**.** *Let $\mathcal{M}$ be a manifold equipped with an (affine) connection ∇. For each smooth curve $c\colon I \to \mathcal{M}$, there exists a unique operator,*

$$\frac{\mathrm{D}}{\mathrm{d}t} : \mathfrak{X}(c) \to \mathfrak{X}(c),$$

called the induced covariant derivative *(induced by ∇), satisfying the following conditions. For all smooth vector fields $Y, Z \in \mathfrak{X}(c)$, $U \in \mathfrak{X}(\mathcal{M})$, $g \in \mathfrak{F}(I)$, and $a, b \in \mathbb{R}$:*

1. *($\mathbb{R}$-linearity) $\frac{\mathrm{D}}{\mathrm{d}t}(aY + bZ) = a\frac{\mathrm{D}}{\mathrm{d}t}Y + b\frac{\mathrm{D}}{\mathrm{d}t}Z$.*
2. *(Leibniz rule) $\frac{\mathrm{D}}{\mathrm{d}t}(gZ) = g'Z + g\frac{\mathrm{D}}{\mathrm{d}t}Z$.*
3. *(Chain rule) $\left(\frac{\mathrm{D}}{\mathrm{d}t}(U \circ c)\right)(t) = \nabla_{\dot{c}(t)}U$, for all $t \in I$.*

If moreover $\mathcal{M}$ is a Riemannian manifold and ∇ is compatible with its metric $\langle\cdot,\cdot\rangle$ (e.g., if ∇ is the Riemannian connection), then the induced covariant derivative $\frac{\mathrm{D}}{\mathrm{d}t}$ also satisfies:

4. *(Product rule) $\frac{\mathrm{d}}{\mathrm{d}t}\langle Y, Z\rangle = \left\langle \frac{\mathrm{D}}{\mathrm{d}t}Y, Z\right\rangle + \left\langle Y, \frac{\mathrm{D}}{\mathrm{d}t}Z\right\rangle$. (see (2.32))*

By the above theorem, any connection ∇ on $\mathcal{M}$ uniquely determines an induced covariant derivative $\frac{\mathrm{D}}{\mathrm{d}t}$. This induced covariant derivative can be defined without any Riemannian metric, as demonstrated by its first three properties. Formally, the connection ∇ assigns a unique operator $\frac{\mathrm{D}}{\mathrm{d}t} : \mathfrak{X}(c) \to \mathfrak{X}(c)$ to each smooth curve c on $\mathcal{M}$.

Definition 24 (Acceleration)**.** Let $c\colon I \to \mathcal{M}$ be a smooth curve. Utilizing the operator $\frac{\mathrm{D}}{\mathrm{d}t}$, the *acceleration* of curve c is defined as

$$\ddot{c}(t) := \frac{\mathrm{D}}{\mathrm{d}t}\dot{c},$$

where $\dot{c}$ is the velocity vector field of c, see (2.31).

Definition 25 (Geodesic)**.** A smooth curve $c\colon I \to \mathcal{M}$ is called a *geodesic* (with respect to ∇) if its acceleration vanishes at every point; that is, $\ddot{c}(t) = 0$ for all t in I.

Example 11. When $\mathcal{M} = \mathbb{R}^n$, for any $x, v \in \mathbb{R}^n$, the curve $\gamma(t) = x + tv$ is a geodesic.

2.11 Exponential Map

This section introduces the exponential map. Exponential map is a kind of (theoretically perfect) retraction R such that the curve it derives, i.e., $t \mapsto \gamma(t) := \mathrm{R}_x(t\xi)$, is precisely the geodesic. However, the exponential map is computationally expensive for many manifolds. Therefore, we often use exponential map for theoretical analysis and other retractions in practical algorithms. Let us define the exponential map starting with the maximal geodesic.

We say that a geodesic $\gamma\colon I \to \mathcal{M}$ is *maximal* if it cannot be extended to a geodesic on a larger interval (its domain), namely, there is no another geodesic $\tilde{\gamma} : \widetilde{I} \to \mathcal{M}$ such that the interval $\tilde{I}$ properly containing I and $\tilde{\gamma}|_I = \gamma$. The next important result comes from [126, Theorem 4.27, Corollary 4.28].

Theorem 5 (Existence and Uniqueness of Maximal Geodesic)**.** *Let $\mathcal{M}$ be a manifold with a connection ∇. For each pair of $(x, v) \in T\mathcal{M}$, there exists a unique maximal geodesic $\gamma\colon I \to \mathcal{M}$ defined on some open interval I containing 0 such that*

$$\gamma(0) = x \quad \textit{and} \quad \dot{\gamma}(0) = v.$$

This unique maximal geodesic γ is determined by the initial point x and initial velocity $v \in T_x\mathcal{M}$, and is therefore denoted by γ_v. For simplicity, we do not specify the base point x (see Remark 5). Thus, the assignment

$$v \mapsto \gamma_v$$

defines a map from $T\mathcal{M}$ to the set of geodesics on $\mathcal{M}$. In particular, when $v = 0_x$, then $\gamma_v(t) \equiv x$ for all $t \in \mathbb{R}$. By [126, Lemma 5.18], for all $v \in T\mathcal{M}$, and $\lambda, t \in \mathbb{R}$, we have $\gamma_{\lambda v}(t) = \gamma_v(\lambda t)$ whenever either side is defined. Now we are ready to define the exponential map.

Definition 26 (Exponential Map)**.** Consider the subset of the tangent bundle:

$$\mathcal{O} = \{(x, v) \in T\mathcal{M} : \gamma_v \text{ is defined on an interval containing } [0, 1]\},$$

and we define the *exponential map*,

$$\mathrm{Exp}\colon \mathcal{O} \to \mathcal{M},\ \mathrm{Exp}(x, v) \equiv \mathrm{Exp}_x(v) := \gamma_v(1).$$

The restriction Exp_x is defined on $\mathcal{O}_x := \mathcal{O} \cap T_x\mathcal{M}$.

Example 12. When $\mathcal{M} = \mathbb{R}^n$, $\mathrm{Exp}_x(v) = x + v$.

Example 13 (Exponential Map on Sphere)**.** Recall Example 3. Let $x \in \mathrm{Sp}^{n-1}$ and $\xi \in T_x\,\mathrm{Sp}^{n-1}$. The exponential map on sphere Sp^{n-1} is given by

$$\mathrm{Exp}_x(\xi) = \cos(\|\xi\|_2)x + \frac{\sin(\|\xi\|_2)}{\|\xi\|_2}\xi.$$

Example 14 (Exponential Map on Stiefel Manifold [72, Corollary 2.2])**.** Recall Example 3. Let $X \in \mathrm{St}(n,k)$ and $\xi \in T_X \mathrm{St}(n,k)$. The exponential map on Stiefel manifold is given by

$$\mathrm{Exp}_X(\xi) = [X, Q] \exp\left(\begin{bmatrix} X^T\xi & -R^T \\ R & 0_{k\times k} \end{bmatrix}\right) \begin{bmatrix} I_k \\ 0_{k\times k} \end{bmatrix},$$

where $QR = (I_n - XX^T)\xi$ is the thin/reduced QR decomposition of $(I_n - XX^T)\xi$ with $Q \in \mathrm{St}(n,k)$ and $R \in \mathbb{R}^{k\times k}$, and $\exp(Z) = \sum_{k=0}^{\infty} \frac{1}{k!} Z^k$ is matrix exponential of square matrix Z.

The next proposition collects some important properties of exponential map. The first three properties are given in [126, Proposition 5.19] and show that an exponential map is a retraction on its domain. Here, to get a more general notion of retraction, we can relax its domain of definition to a subset of tangent bundle $T\mathcal{M}$ (In Definition 15, we ask R to be defined on whole $T\mathcal{M}$). And the last two are given in [64, Lemma 3.5].

Proposition 4 (Properties of Exponential Map)**.** *Let $\mathcal{M}$ be a Riemannian manifold, and let* Exp *be its exponential map.*

(1) $\mathrm{Exp}\colon \mathcal{O} \to \mathcal{M}$ *is a smooth map on its domain $\mathcal{O}$, which is open subset in $T\mathcal{M}$.*

(2) For all $x \in \mathcal{M}$, $\mathrm{Exp}_x(0_x) = x$*; and under the canonical identification $T_{0_x}(T_x\mathcal{M}) \cong T_x\mathcal{M}$, we have* $\mathrm{D}\,\mathrm{Exp}_x(0_x) = \mathrm{Id}_{T_x\mathcal{M}}$.

(3) For all $(x,v) \in T\mathcal{M}$, the maximal geodesic γ_v given in Theorem 5 satisfies $\gamma_v(t) = \mathrm{Exp}_x(tv)$ for all t such that either side is defined.

(4) For all $(x,v) \in \mathcal{O}$ and any $w \in T_x\mathcal{M}$, under the canonical identification $T_v(T_x\mathcal{M}) \cong T_x\mathcal{M}$, we have $\langle \mathrm{D}\,\mathrm{Exp}_x(v)[v], \mathrm{D}\,\mathrm{Exp}_x(v)[w]\rangle = \langle v, w\rangle$.

(5) For all $(x,v) \in \mathcal{O}$, and $\lambda \in [0,1]$, we have $\|\mathrm{D}\,\mathrm{Exp}_x(\lambda v)[v]\| = \|v\|$.

Using exponential map only on a subset $\mathcal{O}$ looks restrictive. In the next section we will show that if $\mathcal{M}$ is complete then $\mathcal{O}$ actually equals the whole $T\mathcal{M}$.

2.12 Metric Space

This section begins with an introduction to the Riemannian distance, which allows us to transform Riemannian manifolds into metric spaces. For details on metric space, see [150, 188]. Importantly, the topology induced by Riemannian distance is identical to the original topology. The concept of completeness for metric space is also closely related to geodesics and exponential map.

2.12.1 Riemannian Distance

A *curve segment* on a manifold $\mathcal{M}$ is a continuous map $c\colon [a,b] \to \mathcal{M}$, and it is *smooth* if and only if it can be extended to a smooth curve $\tilde{c} : I \to \mathcal{M}$ on some open interval I containing $[a,b]$. The term "extend" means that $c(t) = \tilde{c}(t)$ for $t \in [a,b]$. In this case, we let $\dot{c}(a)$ and $\dot{c}(b)$ denote $\dot{\tilde{c}}(a)$ and $\dot{\tilde{c}}(b)$, respectively. Hence, a smooth curve segment has a well-defined velocity $\dot{c}(t) \in T_{c(t)}\mathcal{M}$ for all $t \in [a,b]$.

Furthermore, we say that curve segment c is *regular* if it is smooth with non-vanishing velocity, i.e., $\dot{c}(t) \neq 0$ for all $t \in [a,b]$. Lastly, we describe a curve segment $c\colon [a,b] \to \mathcal{M}$ as *piecewise regular* if it

can be partitioned in a manner, specifically $a = t_0 < t_1 < \ldots < t_{k-1} < t_k = b$, such that its restrictions $c|_{[t_{i-1},t_i]}$ are regular for $i = 1, 2, \ldots, k$. A curve segment $c\colon [a, b] \to \mathcal{M}$ connects x to y if $c(a) = x$ and $c(b) = y$. A manifold $\mathcal{M}$ is *connected* if each pair of points on $\mathcal{M}$ is connected by a curve segment.

Remark 11. Throughout this thesis, in cases where a manifold is not connected, our attention is directed to each of its connected components.

Let $\mathcal{M}$ be a connected Riemannian manifold. Given a piecewise regular curve segment $c\colon [a, b] \to \mathcal{M}$, we define the *length* of c as the integral of its speed (the norm of velocity is called *speed*):

$$L(c) = \int_a^b \|\dot{c}(t)\|_{c(t)} \, \mathrm{d}t.$$

If c is regular, the integrand above is continuous on $[a, b]$ (review (2.21)) and thus integral is well defined [160, Theorem 6.8, Page 125]. If there are "kinks" in the curve, we consider the integrals of separate smooth curve segments and compute the final result by summing over these segments.

The notion of length of a curve leads to a natural distance on $\mathcal{M}$, called the *Riemannian distance*:

$$d(x, y) := \inf\{L(c) \mid c \text{ is a piecewise regular curve segment on } \mathcal{M} \text{ connecting } x \text{ to } y\} \tag{2.33}$$

By [126, Proposition 2.50], if $\mathcal{M}$ is connected then any two points of $\mathcal{M}$ can be joined by a piecewise regular curve segment, and thus (2.33) is well-defined for each $x, y \in \mathcal{M}$. It can be shown that d is a legitimate distance function $\mathcal{M} \times \mathcal{M} \to \mathbb{R}$ satisfying non-negativity, symmetry, and the triangle inequality (see proof of [126, Theorem 2.55]).

Example 15. When $\mathcal{M} = \mathbb{R}^n$, the Riemannian distance is the standard Euclidean distance $d(x, y) = \|x - y\|_2$.

Equipped with such Riemannian distance, $\mathcal{M}$ becomes a metric space and then generate a natural topology, called *metric topology*. Specifically, we define the open ball of radius $r > 0$ centered at a point $x \in \mathcal{M}$ as:

$$B_r(x) = \{y \in \mathcal{M} : d(x, y) < r\}.$$

A subset $\mathcal{U}$ of $\mathcal{M}$ is defined to be open if for each point $x \in \mathcal{U}$, there exists some $r > 0$ such that $B_r(x) \subset \mathcal{U}$. We hope to see this topological structure is the same as the original topological structure (given as a priori in Definition 1), and indeed it is. The following important result comes from [126, Theorem 2.55].

Theorem 6 (Riemannian Manifolds as Metric Spaces)**.** *Let $\mathcal{M}$ be a connected Riemannian manifold. With the distance function given by (2.33), $\mathcal{M}$ is a metric space whose metric topology coincides with its original topology (given as a priori in Definition 1).*

2.12.2 Completeness

Recall that a sequence $\{x_n\}$ in a metric space $\mathcal{M}$ is *Cauchy* if for every $\varepsilon > 0$ there exists an integer k such that $d(x_m, x_n) < \varepsilon$ holds for all $m, n > k$. The theorem below is a fundamental classical result. For a proof, refer to [64, Theorem 2.8, Page 146] or [126, Theorem 6.19].

Theorem 7 (Hopf-Rinow Theorem)**.** *Let $\mathcal{M}$ be a connected Riemannian manifold. Then the following four statements are equivalent:*

(i) (Heine-Borel property) the closed and bounded subsets of $\mathcal{M}$ are compact;

(ii) $\mathcal{M}$ is (metrically) complete *as a metric space equipped with a Riemannian distance, i.e., every Cauchy sequence converges on the manifold.*

(iii) $\mathcal{M}$ is geodesically complete*, i.e., every geodesic on the manifold can be extended to a geodesic defined on the whole $\mathbb{R}$.*

(iv) The domain of the exponential map $\mathcal{O}$ is the whole tangent bundle $T\mathcal{M}$. That is, for every $x \in \mathcal{M}$, the exponential map Exp_x is defined on the entire tangent space $T_x\mathcal{M}$.

In addition, any of the statements above implies that

(v) each pair of $x, y \in \mathcal{M}$ can be joined by a (not necessarily unique) minimal geodesic segment $c\colon [0,1] \to \mathcal{M}$, i.e., $c(0) = x, c(1) = y$ and $d(x,y) = L(c)$.

By Hopf-Rinow Theorem, we conclude that the bounded subset in a complete manifold $\mathcal{M}$ lies in a compact subset of $\mathcal{M}$.

Remark 12. Throughout the thesis, we assume that $\mathcal{M}$ is a connected, complete Riemannian manifold. In case of (v) above, if a unique geodesic segment joining p to q exists, we denote it by γ_{pq}. We will often omit the subscript if it is clear.

As a result of the following conclusions, complete manifolds are quite common, e.g., Stiefel manifold.

Lemma 1 ([34, Example 10.10])**.** *Compact Riemannian manifold is complete.*

2.13 Parallel Transport

Tangent vectors at different points on a manifold $\mathcal{M}$ belong to different tangent spaces, and therefore cannot be directly added together or compared. In differential geometry, there are a few ways to compare vectors in different tangent spaces, and parallel transport is one of the most fundamental methods. Parallel transport provides a way to move a tangent vector from one point to another along a curve in a way that the vector remains "parallel" to itself. This section intrudes the parallel transport.

Consider a manifold $\mathcal{M}$ equipped with a connection ∇, thus with the induced covariant derivative $\frac{\mathrm{D}}{\mathrm{d}t}$. Let $c\colon I \to \mathcal{M}$ be a smooth curve. Recall that $\mathfrak{X}(c)$ denotes the set of all smooth vector fields along the curve c, see (2.29). We say that a vector field $Z \in \mathfrak{X}(c)$ is *parallel* if $\frac{\mathrm{D}}{\mathrm{d}t} Z = 0$. The next important result comes from [126, Theorem 4.32].

Proposition 5 (Existence and Uniqueness of Parallel Vector Field)**.** *For any smooth curve $c\colon I \to \mathcal{M}, t_0 \in I$ and $\xi \in \mathrm{T}_{c(t_0)}\mathcal{M}$, there exists a unique parallel vector field $Z \in \mathfrak{X}(c)$ such that $Z(t_0) = \xi$.*

Definition 27 (Parallel Transport)**.** Given a smooth curve γ on $\mathcal{M}$, *parallel transport* of the tangent space at $\gamma(t_0)$ to the tangent space at $\gamma(t_1)$ along γ,

$$\mathrm{P}_{\gamma}^{t_1 \to t_0} : T_{\gamma(t_0)}\mathcal{M} \to T_{\gamma(t_1)}\mathcal{M}$$

is defined by $\mathrm{P}_{\gamma}^{t_1 \to t_0}(\xi) = Z(t_1)$, where Z is the unique parallel vector field such that $Z(t_0) = \xi$.

Note that the parallel transport from x to y depends on the curve γ connecting x and y. The parallel transport $\mathrm{P}_{\gamma}^{t_1 \to t_0}$ is linear; $\mathrm{P}_{\gamma}^{t_2 \to t_1} \circ \mathrm{P}_{\gamma}^{t_1 \to t_0} = \mathrm{P}_{\gamma}^{t_2 \to t_0}$; $\mathrm{P}_{\gamma}^{t \to t}$ is the identity. In particular, the inverse of $\mathrm{P}_{\gamma}^{t_1 \to t_0}$ is $\mathrm{P}_{\gamma}^{t_0 \to t_1}$.

Example 16. When $\mathcal{M} = \mathbb{R}^n$, the parallel transport of a vector v from a point x to another point y along any curve is the identity map. This is because the tangent spaces of Euclidean spaces do not change. They remain the same, see (2.9).

2.14 Vector Transports

In last section, we have introduce a tool called parallel transport in order to move a tangent vector from one tangent space to another. However, parallel transport may be computationally expensive or even intractable for certain types of manifolds. For example, on the Stiefel manifold, no explicit formula for parallel transport is known, making it difficult to apply in practical algorithm. Now, in this section let us move on to the concept of a vector transport T on a manifold $\mathcal{M}$. It is a more general concept that includes parallel transport as a special case.

Roughly speaking, a vector transport takes in a pair of tangent vectors in the same tangent space and returns another tangent vector in the other tangent space. Define the *Whitney sum* of tangent bundles:

$$T\mathcal{M} \oplus T\mathcal{M} := \{(x, \eta, \xi) : x \in \mathcal{M}, \eta, \xi \in T_x\mathcal{M}\},$$

which can be endowed with a smooth structure. Now we define the vector transport as follows.

Definition 28 (Vector Transports)**.** A smooth map

$$\mathrm{T}\colon T\mathcal{M} \oplus T\mathcal{M} \to T\mathcal{M}\colon (x, \eta, \xi) \mapsto \mathrm{T}_{\eta}(\xi),$$

is called a *vector transport* on $\mathcal{M}$ if there exists a associated retraction R on $\mathcal{M}$ such that T satisfies the following properties for all $x \in \mathcal{M}$:

1. (Associated retraction) $\mathrm{T}_{\eta}(\xi) \in T_{\mathrm{R}_x(\eta)}\mathcal{M}$ for all $\eta, \xi \in T_x\mathcal{M}$.
2. (Consistency) $\mathrm{T}_{0_x}(\xi) = \xi$ for all $\xi \in T_x\mathcal{M}$.
3. (Linearity) $\mathrm{T}_{\eta}(a\xi + b\zeta) = a\mathrm{T}_{\eta}(\xi) + b\mathrm{T}_{\eta}(\zeta)$ for all $a, b \in \mathbb{R}$ and $\eta, \xi, \zeta \in T_x\mathcal{M}$.

The first and third conditions establish that fixing any $x \in \mathcal{M}$ and $\eta \in T_x\mathcal{M}$, the map

$$\mathrm{T}_{\eta}\colon T_x\mathcal{M} \to T_{\mathrm{R}_x(\eta)}\mathcal{M},\ \xi \mapsto \mathrm{T}_{\eta}(\xi),$$

is a linear operator. According to the second condition, $\mathrm{T}_{0_x} = \mathrm{Id}_{T_x\mathcal{M}}$. It makes sense because if we try to move a tangent vector to its original tangent space, we should do nothing. Additionally, T is *isometric* if the following equation holds, for all $x \in \mathcal{M}$ and all $\xi, \zeta, \eta \in T_x\mathcal{M}$:

$$\langle \mathrm{T}_{\eta}\xi, \mathrm{T}_{\eta}\zeta \rangle = \langle \xi, \zeta \rangle.$$

In other words, the adjoint and the inverse coincide, i.e., $\mathrm{T}_{\eta}^{*} = \mathrm{T}_{\eta}^{-1}$ for any $x \in \mathcal{M}$ and $\eta \in T_x\mathcal{M}$.

Remark 13. Isometry plays an irreplaceable role in the proofs about quasi-Newton RIPM in Chapter 7. For the submanifolds $\mathcal{M}$ of a linear space $\mathcal{E}$, which are mostly what we consider in practice, there are many ways to construct an isometric vector transport, see [104, Section 2.3].

There are two important classes of vector transport as follows. Let R be a retraction on $\mathcal{M}$.

Example 17 (Differentiated Retraction [5, Eqaution (8.6)])**.** The differentiated retraction,

$$\mathrm{T}_\eta(\xi) := \mathrm{DR}_x(\eta)[\xi],\ x \in \mathcal{M}, \eta, \xi \in T_x\mathcal{M},$$

is a valid vector transport.

Example 18 (Parallel Transport [5, Eqaution (8.2)])**.** For any $x \in \mathcal{M}, \eta \in T_x\mathcal{M}$,

$$\mathrm{T}_\eta\left(\xi\right) := \mathrm{P}_\gamma^{1\to 0}(\xi)$$

is a valid vector transport, where P_γ denotes the parallel transport along the curve $t \mapsto \gamma(t) := \mathrm{R}_x\left(t\eta\right)$. We often omit the superscript $^{1\to 0}$ if it is clear from the context. In particular, parallel transport is isometric.

2.15 Totally Retractive Neighborhood

In this section, we discuss two neighborhoods: retractive neighborhood and totally retractive neighborhood. These concepts were first proposed in [104] and formally stated in [220]. We begin by stating the classical inverse function theorem on manifolds (see [127, Theorem 4.5]).

Theorem 8 (Inverse Function Theorem on Manifolds)**.** *Let $F\colon \mathcal{M} \to \mathcal{N}$ be a smooth map between two manifolds. If $x \in \mathcal{M}$ is a point such that $\mathrm{D}F(x)$ is invertible, then there exist neighborhoods $\mathcal{U} \subset \mathcal{M}$ of x and $\mathcal{V} \subset \mathcal{N}$ of $F(x)$ such that $F|_U : \mathcal{U} \to \mathcal{V}$ is a diffeomorphism.*

Consider the retraction $\mathrm{R}_x\colon T_x\mathcal{M} \to \mathcal{M}$, by property $\mathrm{DR}_x\left(0_x\right) = \mathrm{Id}_{T_x\mathcal{M}}$ and the inverse function theorem above, there exists a neighborhood $\mathcal{V}$ of 0_x in the tangent space $T_x\mathcal{M}$ such that the restriction

$$\mathrm{R}_x|_{\mathcal{V}} : \mathcal{V} \subset T_x\mathcal{M} \to \mathrm{R}_x(\mathcal{V}) \subset \mathcal{M}$$

is a diffeomorphism; thus, $\mathrm{R}_x^{-1}(y)$ is well defined for all $y \in \mathcal{M}$ sufficiently close to x. In this case, we call $\mathrm{R}_x(\mathcal{V})$ a *retractive neighborhood* of x. The well-known concept of the *normal neighborhood* of x (see [64, Page 70]) is defined similarly, but with consideration solely for $\mathrm{R} = \mathrm{Exp}$.

The following existence theorem gives the definition of a totally retractive neighborhood [220, Theorem 2]. Its proof can be arrived at along the lines of [64, Theorem 3.7, Page 72], which is the same statement as Theorem 9 but restricted to $\mathrm{R} = \mathrm{Exp}$.

Theorem 9 (Existence of Totally Retractive Neighborhood)**.** *Let R be a retraction on $\mathcal{M}$. For any $\bar{x} \in \mathcal{M}$ there exist a neighborhood $\mathcal{W}$ of $\bar{x}$ and a constant $\delta > 0$ such that for every $x \in \mathcal{W}$, the restriction*

$$\mathrm{R}_x|_{B_\delta(0_x)} : B_\delta\left(0_x\right) \subset T_x\mathcal{M} \to \mathrm{R}_x(B_\delta\left(0_x\right)) \subset \mathcal{M}$$

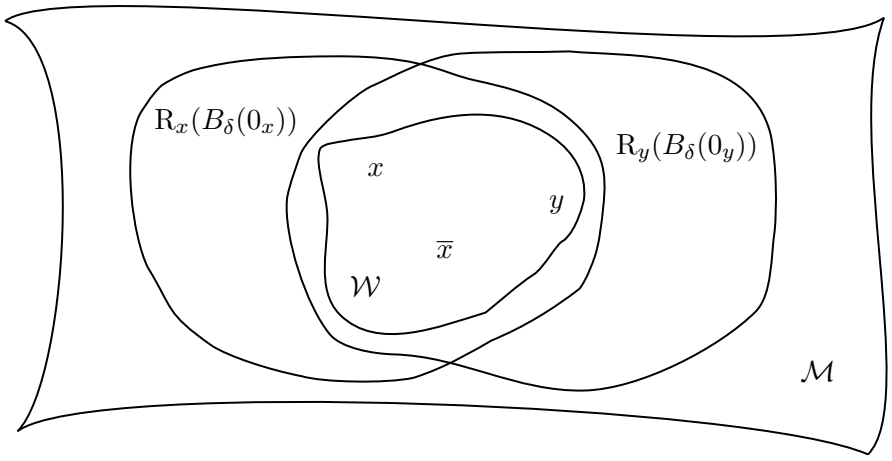


Fig. 2.4 An illustration of totally retractive neighborhood $\bar{x} \in \mathcal{M}$. See Theorem 9 and Remark 14.

is a diffeomorphism and $\mathcal{W} \subset \mathrm{R}_x\left(B_\delta\left(0_x\right)\right)$. *Here* $B_\delta\left(0_x\right)$ *is the open ball in* $T_x\mathcal{M}$ *centered at* 0_x *with radius* δ. *In this case, we call* $\mathcal{W}$ *a* totally retractive neighborhood *of* $\bar{x}$. *See Fig. 2.4.*

Remark 14. The existence of a totally retractive neighborhood shows that there is a neighborhood $\mathcal{W}$ of $\bar{x}$ such that $\mathrm{R}_x^{-1}(y)$ is well defined for any $x, y \in \mathcal{W}$. See Fig. 2.4. In what follows, we will suppose that an appropriate neighborhood has been chosen by default for the well-definedness of $\mathrm{R}_x^{-1}(y)$.

2.16 Lipschitz Continuity with Respect to a Vector Transport

Several Riemannian versions of Lipschitz continuity have been defined, e.g., [34, Section 10.4]. They can all be viewed as Riemannian extension of the usual definition, i.e., they can all reduce to the usual definition of Lipschitz continuity when $\mathcal{M}$ is $\mathbb{R}^n$. However, there are many differences in their formulations. Here, we consider the Lipschitz continuity with respect to a vector transport and its associated retraction. In what follows, let $\mathcal{M}$ be a Riemannian manifold endowed with a vector transport T and an associated retraction R.

2.16.1 Lipschitz Continuity of Gradients and Vector Fields

First, let us consider the Lipschitz-continuous gradient of a scale field f.

Definition 29 ([102, Definition 5.2.1])**.** A function $f\colon \mathcal{M} \to \mathbb{R}$ is *Lipschitz continuously differentiable with respect to* T *in* $\mathcal{U} \subset \mathcal{M}$ if it is differentiable and there exists a number $\kappa > 0$ such that, for all $x, y \in \mathcal{U}$,

$$\|\mathrm{grad}\, f(y) - \mathrm{T}_\eta \,\mathrm{grad}\, f(x)\| \leq \kappa \|\eta\|,$$

where $\eta = \mathrm{R}_x^{-1} y$.

Lemma 2 ([102, Lemma 5.2.1])**.** *If a function* $f\colon \mathcal{M} \to \mathbb{R}$ *is* C^2, *then, for any* $\bar{x} \in \mathcal{M}$ *and for any vector transport* T, *there exists a neighborhood* $\mathcal{U}$ *of* $\bar{x}$ *such that* f *is Lipschitz continuously differentiable with respect to* T *in* $\mathcal{U}$.

We get the following similar results when the gradient vector field is replaced by a general vector field. The proof of Lemma 3 is similar to that of [102, Lemma 5.2.1].

Definition 30. A vector field F on $\mathcal{M}$ is *Lipschitz continuous with respect to* T *in* $\mathcal{U} \subset \mathcal{M}$ if there exists a number $\kappa > 0$ such that, for all $x, y \in \mathcal{U}$,

$$\|F(y) - \mathrm{T}_\eta F(x)\| \leq \kappa \|\eta\|,$$

where $\eta = \mathrm{R}_x^{-1} y$.

Lemma 3. *If F is a C^1 vector field, then, for any $\bar{x} \in \mathcal{M}$ and any vector transport T, there exists a neighborhood $\mathcal{U}$ of $\bar{x}$ such that F is Lipschitz continuous with respect to T in $\mathcal{U}$.*

2.16.2 Lipschitz Continuity of Hessians and Covariant Derivatives

Going one degree higher, let us now discuss the definition of Lipschitz-continuous Hessian of a scale field f. Recall that the Hessian of f associates to each x a linear operator $\mathrm{Hess}\, f(x)$ from and to $T_x\mathcal{M}$. Moreover, the inverse of T_η is needed in the following definitions, so we can assume that vector transport T is isometric, e.g., parallel transport in Example 18 and Remark 13.

Definition 31 ([104, Assumption 3])**.** A function $f\colon \mathcal{M} \to \mathbb{R}$ is *twice Lipschitz continuously differentiable with respect to* T *in* $\mathcal{U} \subset \mathcal{M}$ if it is twice differentiable and there exists a number $\kappa > 0$ such that, for all $x, y \in \mathcal{U}$,

$$\left\|\mathrm{Hess}\, f(y) - \mathrm{T}_\eta \,\mathrm{Hess}\, f(x) \mathrm{T}_\eta^{-1}\right\| \leq \kappa d(x, y),$$

where $\eta = \mathrm{R}_x^{-1} y$.

Lemma 4 ([104, Lemma 4])**.** *If $f\colon \mathcal{M} \to \mathbb{R}$ is C^3, then for any $\bar{x} \in \mathcal{M}$ and any isometric vector transport T, there exists a neighborhood $\mathcal{U}$ of $\bar{x}$, such that f is twice Lipschitz continuously differentiable with respect to T in $\mathcal{U}$.*

If the operator, $\mathrm{Hess}\, f(x)$, above is replaced by a general covariant derivative $\nabla F(x)$, we can get the next results in a similar way. Lemma 5 can be proven in a similar way as [104, Lemma 4].

Definition 32. Given a vector field F on $\mathcal{M}$. The map $x \mapsto \nabla F(x)$ is *Lipschitz continuous with respect to* T *in* $\mathcal{U} \subset \mathcal{M}$ if there exists a number $\kappa > 0$ such that, for all $x, y \in \mathcal{U}$,

$$\left\|\nabla F(y) - \mathrm{T}_\eta \nabla F(x) \mathrm{T}_\eta^{-1}\right\| \leq \kappa d(x, y),$$

where $\eta = \mathrm{R}_x^{-1} y$.

Lemma 5. *If F is a C^2 vector field on $\mathcal{M}$, then for any $\bar{x} \in \mathcal{M}$ and any isometric vector transport T, there exists a neighborhood $\mathcal{U}$ of $\bar{x}$ such that the map $x \mapsto \nabla F(x)$ is Lipschitz continuous with respect to T in $\mathcal{U}$.*

2.17 Local Errors of Retractions and Vector Transports

Notice that in the previous section on the definitions of Lipschitz continuity, we used $\|\eta\|$ with $\eta = \mathrm{R}_x^{-1} y$ or $d(x, y)$ to denote the upper bound on the right-hand side. The next lemma shows that the two are not essentially different. When $\mathcal{M} = \mathbb{R}^n$, both reduce to $\|x - y\|$.

Lemma 6 (Local Errors of Retractions)**.** *Let $\mathcal{M}$ be a Riemannian manifold endowed with a retraction R and let $\bar{x} \in \mathcal{M}$. Then,*

(i) for any $\varepsilon > 0$ there is an $\delta_\varepsilon > 0$ such that for all x in a sufficiently small neighborhood of $\bar{x}$ and all $\xi, \eta \in T_x\mathcal{M}$ with $\|\xi\|, \|\eta\| \leq \delta_\varepsilon$,

$$(1 - \varepsilon)\|\xi - \eta\| \leq d(\mathrm{R}_x(\eta), \mathrm{R}_x(\xi)) \leq (1 + \varepsilon)\|\xi - \eta\|;$$

(ii) there exist $a_0, a_1 > 0$, and $\delta_{a_0,a_1} > 0$ such that for all x in a sufficiently small neighborhood of $\bar{x}$ and all $\xi, \eta \in T_x\mathcal{M}$ with $\|\xi\|, \|\eta\| \leq \delta_{a_0,a_1}$,

$$a_0\|\xi - \eta\| \leq d(\mathrm{R}_x(\eta), \mathrm{R}_x(\xi)) \leq a_1\|\xi - \eta\|;$$

(iii) there exist $a_0, a_1 > 0$, and $\delta_{a_0,a_1} > 0$ such that for all x in a sufficiently small neighborhood of $\bar{x}$ and all $\xi \in T_x\mathcal{M}$ with $\|\xi\| \leq \delta_{a_0,a_1}$,

$$a_0\|\xi\| \leq d(x, \mathrm{R}_x(\xi)) \leq a_1\|\xi\|;$$

(iv) there exist $a_0, a_1 > 0$ such that for all x in a sufficiently small neighborhood of $\bar{x}$,

$$a_0\|\xi\| \leq d(x, \bar{x}) \leq a_1\|\xi\|,$$

where $\xi = \mathrm{R}_{\bar{x}}^{-1}(x)$.

Proof. (i) and (ii) come directly from [157, Lemma 6] and [104, Lemma 2]. Note that if we select any $\varepsilon > 0$ and let $a_0 := 1 - \varepsilon, a_1 := 1 + \varepsilon$, then (i) implies (ii). (iii) follows from (ii) when we take $\eta = 0$. Next, we show (iv). Taking $x = \bar{x}$ in (iii), we have $a_0\|\xi\| \leq d(\bar{x}, \mathrm{R}_{\bar{x}}(\xi)) \leq a_1\|\xi\|$ for all $\xi \in T_{\bar{x}}\mathcal{M}$ with $\|\xi\| \leq \delta_{a_0,a_1}$. Since $\mathrm{R}_{\bar{x}}$ is a local diffeomorphism at $0_{\bar{x}} \in T_{\bar{x}}\mathcal{M}$, for all x in a sufficiently small neighborhood of $\bar{x}$, $\xi = \mathrm{R}_{\bar{x}}^{-1}(x)$ is well defined and $\|\xi\| \leq \delta_{a_0,a_1}$. Therefore, by substituting $\mathrm{R}_{\bar{x}}(\xi) = x$, we have $a_0\|\xi\| \leq d(x, \bar{x}) \leq a_1\|\xi\|$. □

Lemma 7 ([81, Lemma 14.5])**.** *Let F be a C^1 vector field on a Riemannian manifold $\mathcal{M}$ and let $x^* \in \mathcal{M}$ be a nonsingular zero of F, i.e., $F(x^*) = 0$ and $\nabla F(x^*)$ be nonsingular. Then, there exists a neighborhood $\mathcal{U}$ of x^* with constants $a_2, a_3 > 0$ such that, for all $x \in \mathcal{U}$,*

$$a_2 d(x, x^*) \leq \|F(x)\| \leq a_3 d(x, x^*).$$

The following corollary combines Lemma 7 with (iv) of Lemma 6.

Corollary 1. Let F be a C^1 vector field on a Riemannian manifold $\mathcal{M}$ and let $x^* \in \mathcal{M}$ be a nonsingular zero of F. Then, there exists a neighborhood $\mathcal{U}$ of x^* with constants $a_4, a_5 > 0$ such that, for all $x \in \mathcal{U}$,

$$a_4\|\xi\| \leq \|F(x)\| \leq a_5\|\xi\|,$$

where $\xi = \mathrm{R}_{x^*}^{-1}(x)$.

The next lemma compares any two vector transports. In particular, we often compare a certain vector transport with the parallel transport.

Lemma 8 ([104, Lemma 6])**.** *Let $\mathcal{M}$ be a Riemannian manifold endowed with two vector transports T_1 and T_2, and let $\bar{x} \in \mathcal{M}$. Then, there exist a constant a_6 and a neighborhood $\mathcal{U}$ of $\bar{x}$ such that, for all $x, y \in \mathcal{U}$ and all $\xi \in T_y\mathcal{M}$,*

$$\|\mathrm{T}_{1_\eta}^{-1}\xi - \mathrm{T}_{2_\eta}^{-1}\xi\| \leq a_6\|\xi\|\|\eta\|,$$

where $\eta = \mathrm{R}_x^{-1}(y)$.

2.18 Fundamental Theorem of Calculus in Riemannian Case

Let $F\colon \mathbb{R}^n \to \mathbb{R}^n$ be C^1 in an open convex set $D \subset \mathbb{R}^n$ containing x. We know that the fundamental theorem of calculus, e.g., [59, Lemma 4.1.9], is stated as: for any $x + \eta =: y \in D$, we have

$$F(y) - F(x) = \int_0^1 \mathbf{J}_F(x + t\eta)\eta \mathrm{d}t, \tag{2.34}$$

where $\mathbf{J}_F(z) \in \mathbb{R}^{n\times n}$ denotes the Jacobian matrix of F at $z \in \mathbb{R}^n$. In this section, we consider the fundamental theorem of calculus in the Riemannian case.

Lemma 9 ([104, Lemma 8])**.** *Let F be a C^1 vector field on $\mathcal{M}$ endowed with a retraction R and let $\bar{x} \in \mathcal{M}$. Then, there exist a neighborhood $\mathcal{U}$ of $\bar{x}$ and a constant $c_1 \geq 0$ such that, for all $x, y \in \mathcal{U}$,*

$$\left\|\mathrm{P}_\gamma^{0\to 1}F(y) - F(x) - \int_0^1 \mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0}\eta \mathrm{d}t\right\| \leq c_1\|\eta\|^2, \tag{2.35}$$

where $\eta = \mathrm{R}_x^{-1}(y)$ and P_γ is a parallel transport along the curve $t \mapsto \gamma(t) := \mathrm{R}_x(t\eta)$.

Note that if $\mathcal{M} = \mathbb{R}^n$, then (2.35) holds with $c_1 = 0$ and reduces to (2.34), because for $\mathcal{M} = \mathbb{R}^n$, parallel transport P is identity map; $\gamma(t) = x + t\eta$, and linear operator $\nabla F(z)$ identifies matrix $\mathbf{J}_F(z)$. In fact, $c_1 = 0$ also holds for general $\mathcal{M}$ if we choose parallel transport along the geodesic $\gamma(t) = \mathrm{Exp}_x(t\eta)$. This result is also shown in next lemma.

Lemma 10 ([77, Equation (2.4)])**.** *Let F be a C^1 vector field on $\mathcal{M}$ and $\bar{x} \in \mathcal{M}$. Then, there exists a neighborhood $\mathcal{U}$ of $\bar{x}$ such that, for all $x, y \in \mathcal{U}$,*

$$\mathrm{P}_\gamma^{0\to 1}F(y) - F(x) = \int_0^1 \mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0}\eta \mathrm{d}t,$$

where $\eta = \mathrm{Exp}_x^{-1}(y)$ and P_γ is a parallel transport along the geodesic $t \mapsto \gamma(t) := \mathrm{Exp}_x(t\eta)$.

2.19 Some Lemmas on Newton method

In the Euclidean setting, there are two important estimations (inequalities) for convergence analysis of the Newton and quasi-Newton methods for solving nonlinear equations (see [59, Lemma 4.1.12 & 4.1.15]). Here, let $F\colon \mathbb{R}^n \longrightarrow \mathbb{R}^n$ be C^1 in an open convex set $D \subset \mathbb{R}^n$ containing x, and let Jacobian map $x \mapsto \mathbf{J}_F(x)$ be Lipschitz continuous on D with constant γ. Then, for any $x + p =: y \in D$, we have

$$\|F(y) - F(x) - \mathbf{J}_F(x)p\| \leq \frac{\gamma}{2}\|p\|^2. \tag{2.36}$$

Moreover, for any $v, u \in D$, we have

$$\|F(v) - F(u) - \mathbf{J}_F(x)(v-u)\| \leq \gamma \max\{\|v-x\|, \|u-x\|\}\|v-u\|. \tag{2.37}$$

In this section, we consider these estimations in the Riemannian case.

The next lemma corresponds the first estimation (2.36) in the Riemannian case. This result is used for convergence analysis of the Riemannian Newton method.

Lemma 11. *Let F be a C^2 vector field and $\bar{x} \in \mathcal{M}$. Then, there exist a neighborhood $\mathcal{U}$ of $\bar{x}$ and a constant $c_2 > 0$ such that, for all $x \in \mathcal{U}$,*

$$\left\|\mathrm{P}_\gamma^{0\to 1}F(x) - F(\bar{x}) - \nabla F(\bar{x})\eta\right\| \leq c_2 d^2(\bar{x}, x),$$

where $\eta = R_{\bar{x}}^{-1}x$ and P_γ is a parallel transport along the curve $t \mapsto \gamma(t) := \mathrm{R}_{\bar{x}}(t\eta)$.

Proof. It follows from

$$\begin{aligned}
&\left\|\mathrm{P}_\gamma^{0\to 1}F(x) - F(\bar{x}) - \nabla F(\bar{x})\eta\right\| \\
\leq &\left\|\mathrm{P}_\gamma^{0\to 1}F(x) - F(\bar{x}) - \int_0^1 \mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0}\eta \mathrm{d}t\right\| + \left\|\int_0^1 \mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0}\eta \mathrm{d}t - \nabla F(\bar{x})\eta\right\| \\
\leq &c_1\|\eta\|^2 + \left\|\int_0^1 \left[\mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0} - \nabla F(\bar{x})\right]\eta \mathrm{d}t\right\|. \text{ (by Lemma 9)}
\end{aligned}$$

Let θ be the last term of above, i.e., $\theta := \left\| \int_0^1 \left[\mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0} - \nabla F(\bar{x})\right]\eta \mathrm{d}t\right\|$. Note that

$$\begin{aligned}
\theta &\leq \int_0^1 \left\|\mathrm{P}_\gamma^{0\to t}\nabla F(\gamma(t))\mathrm{P}_\gamma^{t\to 0} - \nabla F(\bar{x})\right\| \|\eta\| \,\mathrm{d}t \\
&\leq \int_0^1 c_0 d(\bar{x}, \mathrm{R}_{\bar{x}}(t\eta))\,\|\eta\|\,\mathrm{d}t \text{ (by Lemma 5 and Lipschitz continuity at } \bar{x}) \\
&\leq \int_0^1 c_0 a_1 t\|\eta\|\,\|\eta\|\,\mathrm{d}t = \frac{1}{2}c_0a_1\|\eta\|^2. \text{ (by (iv) of Lemma 6)}
\end{aligned}$$

Again, by (iv) of Lemma 6, we have

$$\left\|\mathrm{P}_\gamma^{0\to 1}F(x) - F(\bar{x}) - \nabla F(\bar{x})\eta\right\| \leq (c_1 + \frac{1}{2}c_0a_1)\|\eta\|^2 \leq (c_1 + \frac{1}{2}c_0a_1)/a_0^2 d^2(\bar{x}, x).$$

Letting $c_2 := (c_1 + \frac{1}{2}c_0a_1)/a_0^2$ completes the proof. □

The next lemma corresponds the second estimation (2.37) in the Riemannian case. This result is used in the convergence analysis of the Riemannian quasi-Newton method, where the transport of tangent vectors in different tangent spaces is required, so we use general vector transport T instead of parallel transport P to formulate the lemma. This point is different from Lemma 11.

Lemma 12. *Let F be a C^2 vector field and $\bar{x} \in \mathcal{M}$. Then, there exist a neighborhood $\mathcal{U}$ of $\bar{x}$ and a constant $c_3 > 0$ such that, for all $x, y \in \mathcal{U}$:*

(i) if $\xi = \mathrm{R}_x^{-1}(y), \zeta = \mathrm{R}_{\bar{x}}^{-1}(x)$, then

$$\left\| \mathrm{T}_{\xi}^{-1} F(y) - F(x) - \mathrm{T}_{\zeta} \nabla F(\bar{x}) \mathrm{T}_{\zeta}^{-1} \xi \right\| \leq c_3 \|\xi\| \max\{d(x, \bar{x}), d(y, \bar{x})\};$$

(ii) if $\eta = \mathrm{R}_x^{-1}(\bar{x})$, $\eta' = \mathrm{R}_{\bar{x}}^{-1}(x)$, then

$$\left\| \mathrm{T}_{\eta}^{-1} F(\bar{x}) - F(x) - \mathrm{T}_{\eta'} \nabla F(\bar{x}) \mathrm{T}_{\eta'}^{-1} \eta \right\| \leq c_3 \|\eta\| d(x, \bar{x}).$$

Proof. (i) can be proven in the same way as [104, Lemma 12], where F is specified as the Riemannian gradient and ∇F as the Riemannian Hessian. (ii) is a special case of (i): set $y = \bar{x}$, $\xi = \eta$, and $\zeta = \eta'$. □

Finally, we end this section with the following well-known lemmas for Newton methods for solving nonlinear equations. They are natural extensions on Riemannian setting.

Lemma 13 (Banach's Lemma [149, Lemma 2.3.2], [117, Theorem 4, Page 155])**.** *Let $\mathcal{A}$, $\mathcal{B}$ be linear operators on some tangent space $T_x\mathcal{M}$ (indeed, any finite-dimensional vector space). If $\mathcal{A}$ is nonsingular and $\|\mathcal{A}^{-1}\|\|\mathcal{B} - \mathcal{A}\| < 1$, then $\mathcal{B}$ is nonsingular and*

$$\|\mathcal{B}^{-1}\| \leq \frac{\|\mathcal{A}^{-1}\|}{1 - \|\mathcal{A}^{-1}\|\|\mathcal{B} - \mathcal{A}\|}.$$

Here, norms above are operator norms (2.1).

Lemma 14 ([74, Lemma 3.2])**.** *Given a vector field F on $\mathcal{M}$. If the map $p \mapsto \nabla F(p)$ is continuous at p^* and $\nabla F(p^*)$ is nonsingular, then there exist a neighborhood $\mathcal{U}$ of p^* and a constant $\Xi > 0$ such that, for all $p \in \mathcal{U}$, $\nabla F(p)$ is nonsingular and $\left\|\nabla F(p)^{-1}\right\| \leqslant \Xi$.*

2.20 Geometry Tools of Product Manifold $\mathcal{N}$

The (Cartesian) product of two (or, more than two) manifolds, $\mathcal{M}_1 \times \mathcal{M}_2$, itself forms a manifold (see [34, Proposition 3.20, Exercise 8.31]). If one is familiar with the geometry tools on $\mathcal{M}_1$ and $\mathcal{M}_2$ individually, then manipulating their product manifold becomes straightforward. Boumal [34] collected the all geometry tools about product manifolds: see [34, Table 7.2] for a summary. In this section, we will consider a special product manifold $\mathcal{N}$ consisting by a general manifold $\mathcal{M}$ and three Euclidean spaces. $\mathcal{N}$ is a very important product manifold for analyzing the Riemannian interior point method in Chapter 5, Chapter 6 and Chapter 7.

Let $\mathcal{M}$ be a Riemannian manifold with metric $\langle \cdot, \cdot \rangle$ and l and m be positive integers. Let Euclidean spaces $\mathbb{R}^l$ and $\mathbb{R}^m$ equipped with the canonical geometry tools, see Example 2, 8, 10, 11, 15, 5, 12, 16.

Considering the product manifold defined by

$$\mathcal{N} := \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m, \tag{2.38}$$

and taking a point denoted by $w := (x, y, z, s) \in \mathcal{N}$, its associated tangent space is the product of tangent spaces (see [34, Proposition 3.20]):

$$T_w\mathcal{N} \cong T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m. \tag{2.39}$$

In above, we used the canonical identification of Euclidean spaces in (2.9). The tangent bundle of $\mathcal{N}$ is $T\mathcal{N} = T\mathcal{M} \times T\mathbb{R}^l \times T\mathbb{R}^m \times T\mathbb{R}^m$ (see [34, Equation (3.31)]).

For any $w \in \mathcal{N}$ and $\xi = (\xi_x, \xi_y, \xi_z, \xi_s), \zeta = (\zeta_x, \zeta_y, \zeta_z, \zeta_s) \in T_w\mathcal{N}$, then product metric

$$\langle \xi, \zeta \rangle_w := \langle \xi_x, \zeta_x \rangle_x + \xi_y^T \zeta_y + \xi_z^T \zeta_z + \xi_s^T \zeta_s, \tag{2.40}$$

defines a Riemannian metric on $\mathcal{N}$ (see [34, Example 3.57]). Then, the induced norm $\|\xi\|_w := \sqrt{\langle \xi, \xi \rangle_w}$ on $T_w\mathcal{N}$ satisfies

$$\|\xi\|_w^2 = \|\xi_x\|_x^2 + \|\xi_y\|_2^2 + \|\xi_z\|_2^2 + \|\xi_s\|_2^2, \tag{2.41}$$

where $\|\cdot\|_2$ denotes the l_2 norm. With metric (2.40), the Riemannian distance on $\mathcal{N}$ is given as, for any $w_1, w_2 \in \mathcal{N}$,

$$d(w_1, w_2) = \sqrt{d^2(x_1, x_2) + \|y_1 - y_2\|_2^2 + \|z_1 - z_2\|_2^2 + \|s_1 - s_2\|_2^2}, \tag{2.42}$$

where d in right side denotes Riemannian distance on $\mathcal{M}$ (see [34, Exercise 10.14]). We conflate the notations of distance d for $\mathcal{N}$ and $\mathcal{M}$ since they are clear from context.

For any $w \in \mathcal{N}$ and $\xi = (\xi_x, \xi_y, \xi_z, \xi_s) \in T_w\mathcal{N}$, the map

$$\bar{\mathrm{R}}_w(\xi) := (\mathrm{R}_x(\xi_x), y + \xi_y, z + \xi_z, s + \xi_s) \tag{2.43}$$

defines a retraction on $\mathcal{N}$ (see [34, Exercise 3.50]). Moreover, $\bar{\mathrm{R}}$ is the exponential map on $\mathcal{N}$ if R is the exponential map on $\mathcal{M}$ (see [34, Exercise 10.32]). For any $w \in \mathcal{N}$ and $\xi, \zeta \in T_w\mathcal{N}$, the map

$$\bar{\mathrm{T}}_\zeta \xi = \bar{\mathrm{T}}_{(\zeta_x, \zeta_y, \zeta_z, \zeta_s)}(\xi_x, \xi_y, \xi_z, \xi_s) := (\mathrm{T}_{\zeta_x}(\xi_x), \xi_y, \xi_z, \xi_s) \tag{2.44}$$

is a vector transport on $\mathcal{N}$ with an associated retraction $\bar{\mathrm{R}}$ in (2.43) if T is a vector transport on $\mathcal{M}$ with an associated R in the right hand of (2.43) ; equivalently, fixing $\zeta \in T_w\mathcal{N}$, it is the linear operator

$$\bar{\mathrm{T}}_\zeta : T_w\mathcal{N} \to T_{\bar{\mathrm{R}}_w(\zeta)}\mathcal{N}, \ \bar{\mathrm{T}}_\zeta := (\mathrm{T}_{\zeta_x}, \mathrm{Id}, \mathrm{Id}, \mathrm{Id}). \tag{2.45}$$

Here, Id means identity map on $\mathbb{R}^l$ or $\mathbb{R}^m$. If it exists, its inverse

$$\bar{\mathrm{T}}_\zeta^{-1} = (\mathrm{T}_{\zeta_x}^{-1}, \mathrm{Id}, \mathrm{Id}, \mathrm{Id}). \tag{2.46}$$

Note that $\bar{\mathrm{T}}$ is isometric if T is isometric.

The next lemma establishes a way of defining a connection on product manifolds. It can easily be extended to the product manifold consisting more than two manifolds.

Lemma 15 (Product Connection [34, Exercise 5.4 & 5.13])**.** *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two Riemannian manifolds, respectively equipped with Riemannian connections ∇^1 and ∇^2. Consider the product manifold $\mathcal{M} = \mathcal{M}_1 \times \mathcal{M}_2$. Let (u_1, u_2) be tangent to $\mathcal{M}$ at (x_1, x_2). Then, the map $\nabla : T\mathcal{M} \times \mathfrak{X}(\mathcal{M}) \to T\mathcal{M}$ defined by*

$$\nabla_{(u_1,u_2)} (F_1, F_2) = \left(\nabla^1_{u_1} F_1(\cdot, x_2) + \mathrm{D}F_1(x_1, \cdot)(x_2)[u_2], \nabla^2_{u_2} F_2(x_1, \cdot) + \mathrm{D}F_2(\cdot, x_2)(x_1)[u_1]\right)$$

is a Riemannian connection on $\mathcal{M}$; we call it the product connection*. The notation $F_1(\cdot, x_2)$ represents the map obtained from $F_1 : \mathcal{M}_1 \times \mathcal{M}_2 \to T\mathcal{M}_1$ by fixing the second input to x_2. In particular, $F_1(\cdot, x_2)$ is a vector field on $\mathcal{M}_1$, while $F_1(x_1, \cdot)$ is a map from $\mathcal{M}_2$ to $T_{x_1}\mathcal{M}_1$. The map F_2 is understood in the same way.*

Consider $\mathbb{R}^l$ and $\mathbb{R}^m$ equipped with the canonical Euclidean connection (Example 10); then, in the way of Lemma 15 the connection of $\mathcal{M}$ essentially determines the connection of $\mathcal{N}$, which is Riemannian provided that connection of $\mathcal{M}$ is Riemannian. In this thesis, we conflate the notations of Riemannian connection ∇ for $\mathcal{N}$ and $\mathcal{M}$ since they are clear from context.

2.21 Note and References

This chapter presents the basic tools of Riemannian geometry, as well as useful results for subsequent chapters. For a more complete understanding of Riemannian geometry, see [127, 126]; for Riemannian optimization, see [5, 34, 165].

We end this chapter with some discussion about quotient manifolds. Riemannian manifolds are extensions of Euclidean space, and Euclidean space is a special case of Riemannian manifolds. In fact, not all manifolds are subsets of Euclidean spaces. For example, let us consider the Grassmannian manifold, denoted as $\mathrm{Gr}(n, p)$, which is the set of all p-dimensional linear subspaces in $\mathbb{R}^n$. Note that the element of $\mathrm{Gr}(n, p)$ is not a point in $\mathbb{R}^n$. Actually, we can define Grassmannian manifold form some quotient structure; thus, such manifolds are called quotient manifolds. They can also become the Riemannian manifolds with some Riemannian metric. Hence, theoretically, manifolds can be very abstract.

A famous example of an optimization problem over $\mathrm{Gr}(n, p)$ is principal component analysis (PCA). Given k points $y_1, \ldots, y_k \in \mathbb{R}^n$, the goal of PCA is to find a linear subspace $L \in \mathrm{Gr}(n, p)$ which fits the data $y_1, \ldots, y_k$ as well as possible, in the sense that it solves $\min_{L \in \mathrm{Gr}(n,p)} \sum_{i=1}^k (L, y_i)^2$, where $\mathrm{dist}(L, y)$ is the Euclidean distance between y and the point in L closest to y. This particular objective function admits an explicit solution involving the SVD of the data matrix $M = [y_1, \ldots, y_k]$. However, this is not the case for other objective functions. For these, we may need more general optimization algorithms to address:

$$\min_{L \in \mathrm{Gr}(n,p)} f(L)$$

where objective function $f\colon \mathrm{Gr}(n, p) \to \mathbb{R}$. Clearly, Euclidean optimization cannot solve these problems unless we convert the problem into some equivalent Euclidean problem.

Part II

Proposal I - Riemannian Smoothing Method (RSM)

Chapter 3

Riemannian Smoothing Method (RSM)

Part	Section
Part 1. Preparation work	3.1 Review of Smoothing Methods 3.2 Euclidean Generalized Subdifferentials 3.3 Riemannian Generalized Subdifferentials 3.4 Smoothing Functions 3.5 Riemannian Gradient Sub-Consistency
Part 2. Core proposal	3.6 Riemannian Smoothing Method
Part 3. Experiments and notes	3.7 Numerical Experiments 3.8 Note and References

Now let us consider the Nonsmooth Riemannian Optimization (NRO):

$$\min_{x\in\mathcal{M}} f(x), \tag{NRO}$$

where $\mathcal{M} \subset \mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}$ is a lower semi-continuous function on $\mathbb{R}^n$. Indeed, f may be nonsmooth or even non-Lipschitzian. For convenience, the term Smooth Riemannian Optimization (SRO) refers to (NRO) when $f(\cdot)$ is continuously differentiable on $\mathbb{R}^n$. To avoid confusion, we use g instead of f,

$$\min_{x\in\mathcal{M}} g(x). \tag{SRO}$$

Throughout this subsection, we will refer to many of the concepts in [214].

3.1 Review of Smoothing Methods

Smoothing methods [52], which use a parameterized smoothing function to approximate the objective function, are effective on a class of nonsmooth optimizations in Euclidean space. Recently, Zhang, Chen and Ma [214] extended a smoothing steepest descent method to the case of Riemannian submanifolds in $\mathbb{R}^n$. This is not the first time that smoothing methods have been studied on manifolds. Liu and Boumal [128] extended the augmented Lagrangian method and exact penalty method to the Riemannian case. The latter leads to a nonsmooth Riemannian optimization problem to which they applied smoothing

techniques. Cambier and Absil [43] dealt with the problem of robust low-rank matrix completion by solving a Riemannian optimization problem, wherein they applied a smoothing conjugate gradient method.

In this section, we propose a general Riemannian smoothing method and apply it to the CP factorization problem.

3.2 Euclidean Generalized Subdifferentials

First, let us review the usual concepts and properties related to generalized subdifferentials on $\mathbb{R}^n$. The function is termed lower semi-continuous (l.s.c.) if, intuitively, it does not "jump upwards" at any point.

We say that $f\colon \mathbb{R}^n \to \mathbb{R}$ is lower semi-continuous (l.s.c.) at $x \in \mathbb{R}^n$ if for every $\varepsilon > 0$, there exists $\delta > 0$ such that for all $y \in B_\delta(x)$,

$$f(x) - \varepsilon < f(y). \tag{3.1}$$

A function is lower semi-continuous if it is lower semi-continuous at every point in $\mathbb{R}^n$. Similarly, we say that f is upper semi-continuous (u.s.c.) at x if (3.1) is replaced by $f(y) < f(x) + \varepsilon$. It is clear that f is continuous at x if and only if f is lower semi-continuous and upper semi-continuous at this point.

Definition 33 (Subdifferential)**.** For a lower semi-continuous function $f\colon \mathbb{R}^n \to \mathbb{R}$.

- The regular (or Fréchet) subdifferential of f at $x \in \mathbb{R}^n$ are defined as

$$\begin{aligned}\hat{\partial} f(x) := \{\operatorname{egrad} h(x) : \exists \delta > 0 \text{ such that } h \in C^1(B_\delta(x)) \text{ and} \\ f - h \text{ attains a local minimum at } x \text{ on } \mathbb{R}^n\}.\end{aligned} \tag{3.2}$$

- The limiting subdifferential of f at $x \in \mathbb{R}^n$ are defined as

$$\partial f(x) := \{\lim_{\ell \to \infty} v^\ell : v^\ell \in \hat{\partial} f(x^\ell), (x^\ell, f(x^\ell)) \to (x, f(x))\}.$$

The above definition (3.2) of regular (or Fréchet) subdifferential $\hat{\partial} f(x)$ is not the standard one; the standard definition follows [158, 8.3 Definition]. But these two definitions are equivalent by [158, 8.5 Proposition]. For locally Lipschitz functions, the Clarke subdifferential at $x \in \mathbb{R}^n$, $\partial^\circ f(x)$, is the convex hull of the limiting subdifferential. Their relationship is as follows:

$$\hat{\partial} f(x) \subset \partial f(x) \subset \partial^\circ f(x).$$

In particular, if f is convex, $\partial f(x)$ and $\partial^\circ f(x)$ coincide with the classical subdifferential in convex analysis [158, 8.12 Proposition].

Example 19. From a result on the pointwise max-function in convex analysis, we have

$$\partial \max(x) = \operatorname{conv}\{\mathbf{e}_i : i \in \mathbb{A}(x)\},$$

where "conv" denotes the convex hull, $\mathbf{e}_i$'s are the standard bases of $\mathbb{R}^n$ and $\mathbb{A}(x) := \{i : x_i = \max(x)\}$. Note that $\max\colon \mathbb{R}^n \to \mathbb{R}$ is a convex function. For proofs, see [10, Theorem 3.23].

3.3 Riemannian Generalized Subdifferentials

Next, we extend our discussion to include generalized subdifferentials of a nonsmooth function on submanifolds $\mathcal{M}$.

Definition 34 (Riemannian Subdifferential)**.** Let $\mathcal{M}$ be an embedded submanifold of $\mathbb{R}^n$, $x \in \mathcal{M}$, and let $f\colon \mathbb{R}^n \to \mathbb{R}$ be a lower semi-continuous function.

- The Riemannian regular (or Fréchet) subdifferential of f at $x \in \mathcal{M}$ are defined as

$$\hat{\partial}_{\mathrm{R}} f(x) := \{\operatorname{grad} h(x) : \exists \delta > 0 \text{ such that } h \in C^1(B_\delta(x)) \text{ and } f - h \text{ attains a local minimum at } x \text{ on } \mathcal{M}\},$$

- The Riemannian limiting subdifferential of f at $x \in \mathcal{M}$ are defined as

$$\partial_{\mathrm{R}} f(x) := \{\lim_{\ell \to \infty} v^\ell : v^\ell \in \hat{\partial}_{\mathrm{R}} f(x^\ell), (x^\ell, f(x^\ell)) \to (x, f(x))\}.$$

If $\mathcal{M} = \mathbb{R}^n$, the above definitions coincide with the usual regular (or Fréchet) and limiting subdifferentials in $\mathbb{R}^n$.

Proposition 6 ([214, Proposition 1])**.** *Let $\mathcal{M}$ be an embedded submanifold of $\mathbb{R}^n$, $x \in \mathcal{M}$, and $f\colon \mathbb{R}^n \to \mathbb{R}$ be a lower semicontinuous function. Suppose $\mathrm{R} : T\mathcal{M} \to \mathcal{M}$ is a retraction defined in Definition 16. Then*

1. *$\hat{\partial}_R f(x) = \hat{\partial}\left(f \circ \mathrm{R}_x\right)(0_x)$ and $\partial_R f(x) = \partial\left(f \circ \mathrm{R}_x\right)(0_x)$;*
2. *$v \in \hat{\partial}_R f(x)$ if and only if $v \in T_x\mathcal{M}$ and the following holds:*

$$f \circ \mathrm{R}_x(\eta_x) \geq f \circ \mathrm{R}_x(0_x) + \langle v, \eta_x \rangle + o(\|\eta_x\|), \quad \forall \eta_x \in T_x\mathcal{M}$$

Definition 35 (Riemannian Limiting Stationary Point)**.** we call a point $x \in \mathcal{M}$ a Riemannian limiting stationary point of (NRO) if

$$0 \in \partial_{\mathrm{R}} f(x). \tag{3.3}$$

It follows directly that, for all $x \in \mathcal{M}$, one has $\hat{\partial}_{\mathrm{R}} f(x) \subset \partial_{\mathrm{R}} f(x)$. According to Proposition 6, if x is a local minimizer of f on $\mathcal{M}$, then $0 \in \hat{\partial}_{\mathrm{R}} f(x)$. In this thesis, we will treat condition (3.3) as a necessary condition for a local solution of (NRO) to exist.

3.4 Smoothing Functions

The smoothing function is the most important tool of the smoothing method.

Definition 36 (Smoothing Function)**.** A function $\tilde{f}(\cdot,\cdot) : \mathbb{R}^n \times \mathbb{R}_{++} \to \mathbb{R}$ is called a smoothing function of $f\colon \mathbb{R}^n \to \mathbb{R}$, if $\tilde{f}(\cdot, \mu)$ is continuously differentiable in $\mathbb{R}^n$ for any $\mu > 0$,

$$\lim_{z \to x, \mu \downarrow 0} \tilde{f}(z, \mu) = f(x)$$

and there exist a constant $\kappa > 0$ and a function $\omega : \mathbb{R}_{++} \to \mathbb{R}_{++}$ such that

$$|\tilde{f}(x,\mu) - f(x)| \leq \kappa\omega(\mu) \quad \text{with} \quad \lim_{\mu\downarrow 0} \omega(\mu) = 0. \tag{3.4}$$

Example 20 ([53, Lemma 4.4])**.** The log-sum-exp function, $\mathrm{lse}(x,\mu) : \mathbb{R}^n \times \mathbb{R}_{++} \to \mathbb{R}$, given by

$$\mathrm{lse}(x,\mu) := \mu \log\left(\sum_{i=1}^{n} \exp(x_i/\mu)\right),$$

is the smoothing function of $\max(x)$ because we can see that:

(i) $\mathrm{lse}(\cdot,\mu)$ is smooth on $\mathbb{R}^n$ for any $\mu > 0$. Its gradient $\mathrm{egrad}_x\, \mathrm{lse}(x,\mu)$ is given by $\sigma(\cdot,\mu) : \mathbb{R}^n \to \Delta^{n-1} \subset \mathbb{R}^n$,

$$\mathrm{egrad}_x\, \mathrm{lse}(x,\mu) = \sigma(x,\mu) := \frac{1}{\sum_{\ell=1}^{n} \exp(x_\ell/\mu)}[\ \exp(x_1/\mu), \cdots, \exp(x_n/\mu)\ ]^T,$$

where $\Delta^{n-1} := \{x \in \mathbb{R}^n : \sum_{i=1}^{n} x_i = 1, x_i \geq 0\}$ is the unit simplex.

(ii) For all $x \in \mathbb{R}^n$ and $\mu > 0$, we have $\max(x) < \mathrm{lse}(x,\mu) \leq \max(x) + \mu\log(n)$. Then, the constant $\kappa = \log(n)$ and $\omega(\mu) = \mu$. The above inequalities imply that $\lim_{z\to x,\mu\downarrow 0} \mathrm{lse}(z,\mu) = \max(x)$.

Table 3.1 List of smoothing functions of the absolute value function $f(x) = |x|$ with parameters κ and $\omega(\mu)$ in Definition 36.

Function	κ	$\omega(\mu)$
$\tilde{f}_1(x,\mu) = \begin{cases} \lvert x\rvert & \text{if } \lvert x\rvert > \frac{\mu}{2} \\ \frac{x^2}{\mu} + \frac{\mu}{4} & \text{if } \lvert x\rvert \leq \frac{\mu}{2} \end{cases}$	$\frac{1}{4}$	μ
$\tilde{f}_2(x,\mu) = \sqrt{\mu^2 + x^2}$	1	μ
$\tilde{f}_3(x,\mu) = 2\mu\log(1 + \exp(\frac{x}{\mu})) - x$	$2\log(2)$	μ
$\tilde{f}_4(x,\mu) = x\tanh\left(\frac{x}{\mu}\right)$, where $\tanh(z)$ is the hyperbolic tangent function.	1	μ
$\tilde{f}_5(x,\mu) = x\,\mathrm{erf}\left(\frac{x}{\mu}\right)$, where $\mathrm{erf}(z) := \frac{2}{\sqrt{\pi}}\int_0^z \exp(-t^2)\mathrm{d}t$ is the Gauss error function.	$\frac{2}{e\sqrt{\pi}}$	μ

3.5 Riemannian Gradient Sub-Consistency

Gradient sub-consistency or consistency is crucial to showing that any limit point of the Riemannian smoothing method is also a limiting stationary point of (NRO).

Definition 37 (Gradient Sub-consistency on $\mathbb{R}^n$)**.** A smoothing function $\tilde{f}$ of f is said to satisfy gradient sub-consistency on $\mathbb{R}^n$ if, for any $x \in \mathbb{R}^n$,

$$G_{\tilde{f}}(x) \subset \partial f(x), \tag{3.5}$$

where the subdifferential of f associated with $\tilde{f}$ at $x \in \mathbb{R}^n$ is given by

$$G_{\tilde{f}}(x) := \{u \in \mathbb{R}^n : \mathrm{egrad}_x\, \tilde{f}\,(z_k,\mu_k) \to u \text{ for some } z_k \to x, \mu_k \downarrow 0\}.$$

Definition 38 (Gradient Sub-consistency on $\mathcal{M}$)**.** Similarly, $\tilde{f}$ is said to satisfy Riemannian gradient sub-consistency on $\mathcal{M}$ if, for any $x \in \mathcal{M}$,

$$G_{\tilde{f},\mathrm{R}}(x) \subset \partial_{\mathrm{R}} f(x), \tag{3.6}$$

where the Riemannian subdifferential of f associated with $\tilde{f}$ at $x \in \mathcal{M}$ is given by

$$G_{\tilde{f},\mathrm{R}}(x) = \{v \in \mathbb{R}^n : \operatorname{grad} \tilde{f}\,(z_k, \mu_k) \to v \text{ for some } z_k \in \mathcal{M}, z_k \to x, \mu_k \downarrow 0\}.$$

Moreover, if one substitutes the inclusion with equality "=" in (3.5), then $\tilde{f}$ satisfies gradient consistency on $\mathbb{R}^n$, and similarly in (3.6) for $\mathcal{M}$. Thanks to the following useful proposition, we can induce gradient sub-consistency on $\mathcal{M}$ from that on $\mathbb{R}^n$ if f is locally Lipschitz.

Proposition 7 ([214, Theorem 2])**.** *Given an embedded submanifold $\mathcal{M}$ of $\mathbb{R}^n$ and a vector $x \in \mathcal{M}$, let f be a locally Lipschitz function near x on $\mathbb{R}^n$, with $\tilde{f}$ being a smoothing function of f. If the gradient subconsistency of $\tilde{f}$ at x on $\mathbb{R}^n$ holds, then the Riemannian gradient subconsistency of $\tilde{f}$ at x on $\mathcal{M}$ holds.*

The next example illustrates Riemannian gradient sub-consistency on $\mathcal{M}$ for $\mathrm{lse}(x, \mu)$ in Example 20, since any convex function is locally Lipschitz continuous.

Example 21 ([53, Lemma 4.4])**.** The smoothing function $\mathrm{lse}(x, \mu)$ of $\max(x)$ satisfies gradient consistency on $\mathbb{R}^n$. That is, for any $x \in \mathbb{R}^n$,

$$\partial \max(x) = G_{\mathrm{lse}}(x) = \{\lim_{x^k \to x, \mu_k \downarrow 0} \sigma(x^k, \mu_k)\}.$$

Note that the original assertion of [53, Lemma 4.4] is gradient consistency in the Clarke sense, i.e., $\partial^\circ \max(x) = G_{\mathrm{lse}}(x)$.

3.6 Riemannian Smoothing Method

Motivated by the previous papers [43, 128, 214] on smoothing methods and Riemannian manifolds, we propose a general Riemannian smoothing method. Algorithm 3 is the basic framework of this general method.

3.6.1 Basic Riemannian Smoothing Method I

Now let us describe the convergence properties of the basic method. First, let us assume that the function $\tilde{f}(x, \mu_k)$ has a minimizer on $\mathcal{M}$ for each value of μ_k.

Theorem 10. *Suppose that each x^k is an exact global minimizer of (3.7) in Algorithm 3. Then every limit point x^* of the sequence $\{x^k\}$ is a global minimizer of (NRO).*

Proof. Let $\bar{x}$ be a global solution of (NRO), that is,

$$f(\bar{x}) \leq f(x) \quad \text{for all } x \in \mathcal{M}.$$

Algorithm 3: Basic Riemannian Smoothing Method for (NRO)

Input: Nonsmooth objective function f and one of its smoothing function $\tilde{f}$, a (smooth) Riemannian algorithm (called "sub-algorithm" here) for (SRO) and an initial point $x^{-1} \in \mathcal{M}$.

Output: Sequence $\{x^k\} \subset \mathcal{M}$.

Set $k \to 0$, $\theta \in (0,1)$ and $\mu_0 > 0$;

while *stopping criterion not satisfied* **do**

1. Solve
$$x^k := \arg\min_{x \in \mathcal{M}} \tilde{f}(x, \mu_k) \tag{3.7}$$
approximately by using the chosen sub-algorithm, starting at x^{k-1};
2. $\mu_{k+1} \to \theta \mu_k$;
3. $k \to k+1$;

end

From the Definition 36 of the smoothing function, there exist a constant $\kappa > 0$ and a function $\omega : \mathbb{R}_{++} \to \mathbb{R}_{++}$ such that, for all $x \in \mathcal{M}$,

$$-\kappa\omega(\mu) \leq \tilde{f}(x, \mu) - f(x) \leq \kappa\omega(\mu) \tag{3.8}$$

with $\lim_{\mu \downarrow 0} \omega(\mu) = 0$. Substituting x^k and combining with the global solution $\bar{x}$, we have that

$$\tilde{f}(x^k, \mu_k) \geq f(x^k) - \kappa\omega(\mu_k) \geq f(\bar{x}) - \kappa\omega(\mu_k).$$

By rearranging this expression, we obtain

$$-\kappa\omega(\mu_k) \leq \tilde{f}(x^k, \mu_k) - f(\bar{x}). \tag{3.9}$$

Since x^k minimizes $\tilde{f}(x, \mu_k)$ on $\mathcal{M}$ for each μ_k, we have that $\tilde{f}(x^k, \mu_k) \leq \tilde{f}(\bar{x}, \mu_k)$, which leads to

$$\tilde{f}(x^k, \mu_k) - f(\bar{x}) \leq \tilde{f}(\bar{x}, \mu_k) - f(\bar{x}) \leq \kappa\omega(\mu_k). \tag{3.10}$$

The second inequality above follows from (3.8). Combining (3.9) and (3.10), we obtain

$$|\tilde{f}(x^k, \mu_k) - f(\bar{x})| \leq \kappa\omega(\mu_k). \tag{3.11}$$

Now, suppose that x^* is a limit point of $\{x^k\}$, so that there is an infinite subsequence $\mathcal{K}$ such that $\lim_{k \in \mathcal{K}} x^k = x^*$. Note that $x^* \in \mathcal{M}$ because $\mathcal{M}$ is complete. By taking the limit as $k \to \infty, k \in \mathcal{K}$, on both sides of (3.11), again by the definition of the smoothing function, we obtain

$$|f(x^*) - f(\bar{x})| = \lim_{k \in \mathcal{K}} |\tilde{f}(x^k, \mu_k) - f(\bar{x})| \leq \lim_{k \in \mathcal{K}} \kappa\omega(\mu_k) = 0.$$

Thus, it follows that $f(x^*) = f(\bar{x})$. Since $x^* \in \mathcal{M}$ is a point whose objective value is equal to that of the global solution $\bar{x}$, we conclude that x^*, too, is a global solution. □

3.6.2 Basic Riemannian Smoothing Method II

This strong result requires us to find a global minimizer of each subproblem, which, however, cannot always be done. The next result concerns the convergence properties of the sequence $\tilde{f}(x^k, \mu_k)$ under the condition that $\tilde{f}$ has the following additional property.

Definition 39 (Additional Property 1 (AP1): Approximate from Above)**.**

$$0 < \mu_2 < \mu_1 \Longrightarrow \tilde{f}(x, \mu_2) < \tilde{f}(x, \mu_1) \text{ for all } x \in \mathbb{R}^n. \tag{AP1}$$

Example 22. The above property holds for $\mathrm{lse}(x, \mu)$ in Example 20; i.e., we have $\mathrm{lse}(x, \mu_2) < \mathrm{lse}(x, \mu_1)$ on $\mathbb{R}^n$, provided that $0 < \mu_2 < \mu_1$. Note that under the equality,

$$\sum_{l=1}^{n} \exp(x_l/\mu) = \exp\{\mathrm{lse}(x, \mu)/\mu\},$$

the i-th component of $\sigma(x, \mu)$ can be rewritten as

$$\sigma_i(x, \mu) = \exp\{(x_i - \mathrm{lse}(x, \mu))/\mu\}.$$

For any fixed $x \in \mathbb{R}^n$, consider the derivative of a real function $\mu \to \mathrm{lse}(x, \cdot) : \mathbb{R}_{++} \to \mathbb{R}$. Then we have

$$\begin{aligned} \mathrm{egrad}_\mu \,\mathrm{lse}(x, \mu) = \mathrm{lse}\,/\mu - \frac{\sum_{i=1}^n x_i \exp(x_i/\mu)}{\mu \exp\left(\mathrm{lse}\,/\mu\right)} &= (\mathrm{lse} - \sum_{i=1}^{n} x_i \exp\{(x_i - \mathrm{lse})/\mu\})/\mu \\ &= (\mathrm{lse} - \sum_{i=1}^{n} x_i \sigma_i)/\mu \leq 0, \end{aligned}$$

where "lse, σ" are shorthand for $\mathrm{lse}(x, \mu)$ and $\sigma(x, \mu)$. For the last inequality above, we observe that $\sigma \in \Delta^{n-1}$; hence, the term $\sum_{i=1}^n x_i \sigma_i$ is a convex combination of all entries of x, which implies that $\sum_{i=1}^n x_i \sigma_i \leq \max(x) < \mathrm{lse}$. This completes the proofs of our claims.

In [43], the authors considered a special case of Algorithm 3, wherein the smoothing function $\tilde{f}(x, \mu) = \sqrt{\mu^2 + x^2}$ of $|x|$ also satisfies (AP1) and a Riemannian conjugate gradient method is used for (3.7).

Theorem 11. *Suppose that $f^* := \inf_{x \in \mathcal{M}} f(x)$ exists and the smoothing function $\tilde{f}$ has property (AP1). Let $f^k := \tilde{f}(x^k, \mu_k)$. Then the sequence $\{f^k\}$ generated by Algorithm 3 is strictly decreasing and bounded below by f^*; hence,*

$$\lim_{k \to \infty} |f^k - f^{k-1}| = 0.$$

Proof. For each $k \geq 1$, x_k is obtained by approximately solving

$$\min_{x \in \mathcal{M}} \tilde{f}(x, \mu_k),$$

starting at x^{k-1}. Then at least, we have

$$\tilde{f}(x^{k-1}, \mu_k) \geq \tilde{f}(x^k, \mu_k) = f^k.$$

Since $\mu_k = \theta\mu_{k-1} < \mu_{k-1}$, property (AP1) ensures

$$f^{k-1} = \tilde{f}(x^{k-1}, \mu_{k-1}) > \tilde{f}(x^{k-1}, \mu_k).$$

The claim that sequence $\{f^k\}$ is strictly decreasing follows from these two inequalities.

Suppose that, for all $\mu > 0$ and for all $x \in \mathbb{R}^n$,

$$\tilde{f}(x, \mu) \geq f(x). \tag{3.12}$$

Then for each k,

$$f^k = \tilde{f}(x^k, \mu_k) \geq f(x^k) \geq \inf_{x\in\mathcal{M}} f(x) = f^*,$$

which proves our claims.

Now, we show (3.12) is true if the smoothing function has property (AP1). Fix any $x \in \mathbb{R}^n$; (AP1) implies that $\tilde{f}(x, \cdot)$ is strictly decreasing as $\mu \to 0$. Actually, $\tilde{f}(x, \cdot)$ is monotonically increasing on $\mu > 0$. On the other hand, from the definition of the smoothing function, we have that

$$\lim_{\mu\downarrow 0} \tilde{f}(x, \mu) = f(x).$$

Hence, we have $\inf_{\mu>0} \tilde{f}(x, \mu) = f(x)$, as claimed. □

3.6.3 Enhanced Riemannian Smoothing Method

Note that the above weak result does not ensure that $\{f^k\} \to f^*$. Next, for better convergence (compared with Theorem 11) and an effortless implementation (compared with Theorem 10), we propose an enhanced Riemannian smoothing method: Algorithm 4. This is closer to the version in [214], where the authors use the Riemannian steepest descent method for solving the smoothed problem (3.13).

Algorithm 4: Enhanced Riemannian Smoothing Method for (NRO)

Input: Nonsmooth objective function f and one of its smoothing function $\tilde{f}$, a (smooth) Riemannian algorithm (called "sub-algorithm" here) for (SRO) and an initial point $x^{-1} \in \mathcal{M}$. A nonnegative sequence $\{\delta_k\} \to 0$.

Output: Sequence $\{x_k\} \subset \mathcal{M}$.

Set $k \to 0$, $\theta \in (0, 1)$ and $\mu_0 > 0$;

while *stopping criterion not satisfied* **do**

1. Solve
$$x^k := \arg\min_{x\in\mathcal{M}} \tilde{f}(x, \mu_k) \tag{3.13}$$
approximately by using the chosen sub-algorithm, starting at x^{k-1}, such that
$$\|\operatorname{grad} \tilde{f}(x^k, \mu_k)\| < \delta_k; \tag{3.14}$$
2. $\mu_{k+1} \to \theta\mu_k$;
3. $k \to k + 1$;

end

The following result is adapted from [214, Proposition 4.2, Theorem 4.3]. Readers are encouraged to refer to [214] for a discussion on the stationary point associated with $\tilde{f}$ on $\mathcal{M}$.

Theorem 12. *In Algorithm 4, suppose that the chosen sub-algorithm has the following general convergence property for (SRO):*

$$\liminf_{\ell\to\infty} \|\operatorname{grad} g(x^\ell)\| = 0. \tag{3.15}$$

Moreover, suppose that, for all μ_k, the function $\tilde{f}(\cdot, \mu_k)$ satisfies the convergence assumptions of the sub-algorithm needed for g above and $\tilde{f}$ satisfies the Riemannian gradient sub-consistency on $\mathcal{M}$ (see Definition 38). Then

1. *For each k, there exists an x^k satisfying (3.14); hence, Algorithm 4 is well-defined.*
2. *Every limit point x^* of the sequence $\{x^k\}$ generated by Algorithm 4 is a Riemannian limiting stationary point of (NRO) (see (3.3)).*

Proof. Fix any μ_k. By (3.15), we have $\liminf_{\ell\to\infty} \|\operatorname{grad} \tilde{f}(x^\ell, \mu_k)\| = 0$. Hence, there is a convergent subsequence of $\|\operatorname{grad} \tilde{f}(x^\ell, \mu_k)\|$ whose limit is 0. This means that, for any $\varepsilon > 0$, there exists an integer ℓ_ε such that $\|\operatorname{grad} \tilde{f}(x^{\ell_\varepsilon}, \mu_k)\| < \varepsilon$. If $\varepsilon = \delta_k$, we get $x^k = x^{\ell_\varepsilon}$. Thus, statement (1) holds.

Next, suppose that x^* is a limit point of $\{x^k\}$ generated by Algorithm 4, so that there is an infinite subsequence $\mathcal{K}$ such that $\lim_{k\in\mathcal{K}} x^k = x^*$. From (1), we have

$$\lim_{k\in\mathcal{K}} \|\operatorname{grad} \tilde{f}(x^k, \mu_k)\| \le \lim_{k\in\mathcal{K}} \delta_k = 0,$$

and we find that $\operatorname{grad} \tilde{f}(x^k, \mu_k) \to 0$ for $k \in \mathcal{K}, x^k \in \mathcal{M}, x^k \to x^*, \mu_k \downarrow 0$. Hence,

$$0 \in G_{\tilde{f},\mathrm{R}}(x^*) \subset \partial_{\mathrm{R}} f(x^*).$$

□

Now let us consider the selection strategy of the nonnegative sequence $\{\delta_k\}$ with $\delta_k \to 0$. In [214], when $\mu_{k+1} = \theta\mu_k$ shrinks, the authors set

$$\delta_{k+1} := \rho\delta_k \tag{3.16}$$

with an initial value of δ_0 and constant $\rho \in (0, 1)$. In the spirit of the usual smoothing methods described in [52], one can set

$$\delta_k := \gamma\mu_k \tag{3.17}$$

with a constant $\gamma > 0$. The latter is an adaptive rule, because μ_k determines subproblem (3.13) and its stopping criterion at the same time. The merits and drawbacks of the two rules require more discussion, but the latter seems to be more reasonable.

3.7 Numerical Experiments

As described at the end of Section 3.6, the algorithms in [214] and [43] are both special cases of our algorithm. In this section, we compare them to show whether it performs better when we use other

sub-algorithms or other smoothing functions. We applied Algorithm 4 to two problems: finding a sparse vector (FSV) in a subspace and robust low-rank matrix completion, which are problems implemented in [214] and [43], respectively. Since they both involve approximations to the ℓ_1 norm, we applied the smoothing functions listed in Table 3.1. We used the 6 solvers built into Manopt 7.0, namely,

- steepest descent;
- Barzilai-Borwein (i.e., gradient-descent with BB step size);
- Conjugate gradient;
- trust regions;
- BFGS (a limited-memory version);
- ARC (i.e., adaptive regularization by cubics).

3.7.1 Finding the Sparsest Vector (FSV)

The first experiment examines the problem of finding the sparsest vector in an n-dimensional linear subspace of $\mathbb{R}^m$ in (FSV). Our synthetic problems of the ℓ_1 minimization model (FSV) were generated in the same way as in [214]: i.e., we chose $m \in \{4n, 6n, 8n, 10n\}$ for $n = 5$ and $m \in \{6n, 8n, 10n, 12n\}$ for $n = 10$. We defined a sparse vector $\mathfrak{e}_n := [1, 2, \ldots, 1, 0, \ldots, 0]^T \in \mathbb{R}^m$, whose first n components are 1 and the remaining components are 0. Let the subspace W be the span of $\mathfrak{e}_n$ and some $n-1$ random vectors in $\mathbb{R}^m$. By `mgson.m` [50], we generated an orthonormal basis of W to form a matrix $Q \in \mathbb{R}^{m\times n}$. With this construction, the minimum value of $\|Qx\|_0$ should be equal to n. We chose the initial points by using the `M.rand()` tool of Manopt 7.0 that returns a random point on the manifold `M` and set `x0 = abs(M.rand())`. The nonnegative initial point seemed to be better in the experiment. Regarding the the settings of our Algorithm 4, we chose the same smoothing function $\tilde{f}_1(x, \mu)$ in Table 3.1 and the same gradient tolerance strategy (3.16) as in [214]: $\mu_0 = 1, \theta = 0.5, \delta_0 = 0.1, \rho = 0.5$. We compared the numerical performances when using different sub-algorithms. Note that with the choice of the steepest-descent method, our Algorithm 4 is exactly the same as the one in [214].

For each (n, m), we generated 50 pairs of random instances and random initial points. We claim that an algorithm successfully terminates if $\|Qx_k\|_0 = n$, where x_k is the k-th iteration. Here, when we count the number of nonzeros of Qx_k, we truncated the entries as

$$(Qx_k)_i = 0 \ \text{ if } \ |(Qx_k)_i| < \tau,$$

where $\tau > 0$ is a tolerance related to the precision of the solution, taking values from 10^{-5} to 10^{-12}. Tables 3.2 and 3.3 report the number of successful cases out of 50 cases. Boldface highlights the best result for each row.

As shown in Table 3.2 and 3.3, surprisingly, the conjugate-gradient method, which performed best on the CP factorization problem in Section 4.6, performed worst on the FSV problem. In fact, it was almost useless. Moreover, although the steepest-descent method employed in [214] was not bad at obtaining low-precision solutions with $\tau \in \{10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}\}$, it had difficulty obtaining high-precision solutions with $\tau \in \{10^{-9}, 10^{-10}, 10^{-11}, 10^{-12}\}$. The remaining four sub-algorithms easily obtained high-precision solutions, with the Barzilai-Borwein method performing the best in most occasions. Combined with the results in Section 4.6, this shows that in practice, the choice of sub-algorithm in the Riemannian

Table 3.2 Number of successes from 50 pairs of random instances and random initial points for the ℓ_1 minimization model (FSV) and $n = 5$.

(n, m)	τ	Steepest-descent	Barzilai-Borwein	Conjugate-gradient	Trust-regions	BFGS	ARC
$(5, 20)$	10^{-5}	21	19	0	22	**23**	**23**
	10^{-6}	21	19	0	22	**23**	**23**
	10^{-7}	21	19	0	22	**23**	**23**
	10^{-8}	16	19	0	22	**23**	**23**
$(5, 30)$	10^{-5}	36	**42**	0	34	36	35
	10^{-6}	36	**42**	0	34	36	35
	10^{-7}	36	**42**	0	34	36	35
	10^{-8}	34	**42**	0	34	36	35
$(5, 40)$	10^{-5}	44	**47**	1	44	**47**	45
	10^{-6}	44	**47**	0	44	**47**	45
	10^{-7}	44	**47**	0	44	**47**	45
	10^{-8}	43	**47**	0	44	**47**	45
$(5, 50)$	10^{-5}	**47**	**47**	2	45	45	45
	10^{-6}	**47**	**47**	2	45	45	45
	10^{-7}	**47**	**47**	0	45	45	45
	10^{-8}	46	**47**	0	45	45	45
(n, m)	τ	Steepest-descent	Barzilai-Borwein	Conjugate-gradient	Trust-regions	BFGS	ARC
$(5, 20)$	10^{-9}	0	19	0	22	**23**	**23**
	10^{-10}	0	19	0	22	**23**	**23**
	10^{-11}	0	19	0	22	**23**	19
	10^{-12}	0	18	0	**22**	**22**	17
$(5, 30)$	10^{-9}	8	**42**	0	34	36	35
	10^{-10}	1	**42**	0	34	36	35
	10^{-11}	0	**42**	0	34	36	33
	10^{-12}	0	**42**	0	34	34	29
$(5, 40)$	10^{-9}	3	**47**	0	44	**47**	45
	10^{-10}	2	**47**	0	44	**47**	45
	10^{-11}	1	**47**	0	44	**47**	44
	10^{-12}	0	**46**	0	44	44	36
$(5, 50)$	10^{-9}	5	**47**	0	45	45	45
	10^{-10}	2	**47**	0	45	45	45
	10^{-11}	0	**47**	0	45	45	45
	10^{-12}	0	**47**	0	45	45	37

smoothing method (Algorithm 4) is highly problem-dependent. For the other smoothing functions in Table 3.1, we obtained similar results as in Table 3.2 and 3.3.

3.7.2 Robust Low-Rank Matrix Completion

The second experiment examines the problem of robust low-rank matrix completion in (RMC). In all of the experiments, the problems were generated in the same way as in [43]. In particular, after picking the values of m, n, r, we generated the ground truth $U \in \mathbb{R}^{m\times r}$, $V \in \mathbb{R}^{n\times r}$ with independent and identically distributed (i.i.d.) Gaussian entries of zero mean and unit variance and $M := UV^T$. We then sampled $k := \rho r(m + n - r)$ observed entries uniformly at random, where ρ is the oversampling factor. In our experiments, we set $\rho = 5$ throughout. We chose the initial points X_0 by using the rank-r truncated SVD decomposition of $P_\Omega(M)$.

Regarding the setting of our Algorithm 4, we tested all combinations of the five smoothing functions in Table 3.1 and six sub-algorithms mentioned before (30 cases in total). We set $\mu_0 = 100$ and chose an aggressive value of $\theta = 0.05$ for reducing μ, as in [43]. The stopping criterion of the loop of the sub-algorithm was set to a maximum of 40 iterations or the gradient tolerance (3.17), whichever was reached first. We monitored the iterations X_k through the Root Mean Square Error (RMSE), which is defined as the error on all the entries between X_k and the original low-rank matrix M_0, i.e.,

$$\text{RMSE}\,(X_k, M_0) := \sqrt{\frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left(X_{k,ij} - M_{0,ij}\right)^2}{mn}}.$$

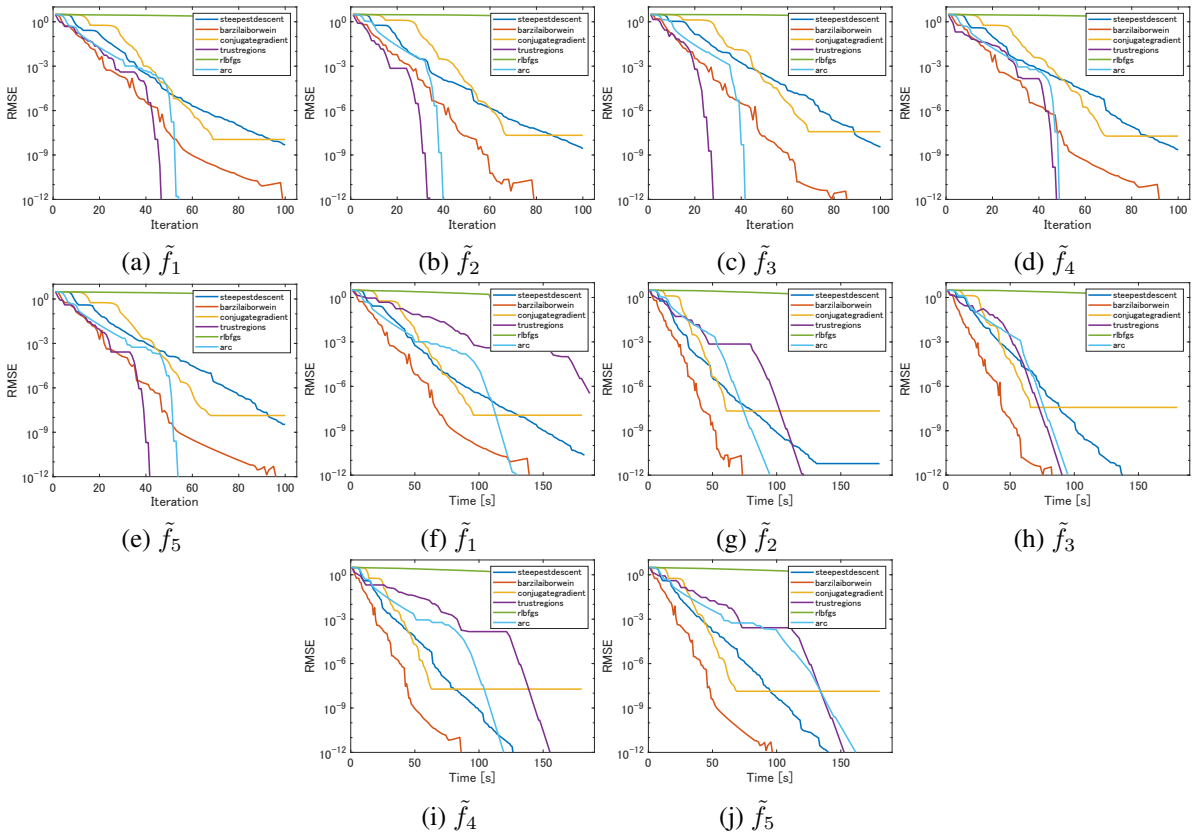
Perfect low-rank matrix completion As in [43], we first tested all the methods on a simple perfect matrix M (without any outliers) of size 5000×5000 and rank 10. The results are shown in Fig. 3.1. We can see that the choice of smoothing function does not have much effect on numerical performance. In terms of the number of iterations ((a)–(e)), our Algorithm 4 inherits the convergence of its sub-algorithm at least Q-superlinearly when trust regions or ARC are used. But the single iteration cost of trust regions and ARC is high; they are not efficient in terms of time. Specifically, the conjugate-gradient method employed in [43] stagnates at lower precision. Overall, Barzilai-Borwein performed best in terms of time and accuracy.

Low-rank matrix completion with outliers Given a 500×500 matrix for which we observed the entries uniformly at random with an oversampling ρ of 5, we perturbed 5% of the observed entries by adding noise to them in order to create outliers. The added item was a random variable defined as $\mathrm{O} = S_{\pm 1} \cdot \mathcal{N}(\mu_N, \sigma_N^2)$ where $S_{\pm 1}$ is a random variable with equal probability of being equal to $+1$ or -1, while $\mathcal{N}(\mu_N, \sigma_N^2)$ is a Gaussian random variable of mean μ_N and variance σ_N^2.

Fig. 3.2 reports the results of two 500×500 instances with outliers generated using $\mu_N = \sigma_N = 0.1$ and $\mu_N = \sigma_N = 1$. Again, we can see that the choice of smoothing function does not have much effect. In most cases, BFGS and trust regions were better than the other methods in terms of number of iterations, and BFGS was the fastest. Furthermore, the conjugate-gradient method employed in [43] still stagnated at solutions with lower precision, approximately 10^{-6}, while steepest descent, BFGS, and trust regions always obtained solutions with at least 10^{-8} precision.

Table 3.3 Number of successes from 50 pairs of random instances and random initial points for the ℓ_1 minimization model (FSV) and $n = 10$.

(n, m)	τ	Steepest-descent	Barzilai-Borwein	Conjugate-gradient	Trust-regions	BFGS	ARC
$(10, 60)$	10^{-5}	24	**28**	0	**28**	**28**	25
	10^{-6}	24	**28**	0	**28**	**28**	25
	10^{-7}	24	**28**	0	**28**	**28**	25
	10^{-8}	23	**28**	0	**28**	**28**	25
$(10, 80)$	10^{-5}	39	37	1	**40**	39	**40**
	10^{-6}	39	37	0	**40**	39	**40**
	10^{-7}	39	37	0	**40**	39	**40**
	10^{-8}	39	37	0	**40**	39	**40**
$(10, 100)$	10^{-5}	45	**48**	3	45	43	41
	10^{-6}	45	**48**	0	45	43	41
	10^{-7}	45	**48**	0	45	43	41
	10^{-8}	45	**48**	0	45	43	41
$(10, 120)$	10^{-5}	44	**46**	1	44	44	43
	10^{-6}	44	**46**	0	44	44	43
	10^{-7}	44	**46**	0	44	44	43
	10^{-8}	44	**46**	0	44	44	43
(n, m)	τ	Steepest-descent	Barzilai-Borwein	Conjugate-gradient	Trust-regions	BFGS	ARC
$(10, 60)$	10^{-9}	3	**28**	0	**28**	**28**	25
	10^{-10}	0	**28**	0	**28**	**28**	25
	10^{-11}	0	**28**	0	**28**	**28**	22
	10^{-12}	0	**28**	0	**28**	27	12
$(10, 80)$	10^{-9}	5	37	0	**40**	39	**40**
	10^{-10}	0	37	0	**40**	39	**40**
	10^{-11}	0	37	0	**40**	39	39
	10^{-12}	0	37	0	**40**	37	30
$(10, 100)$	10^{-9}	13	**48**	0	45	43	41
	10^{-10}	2	**48**	0	45	43	41
	10^{-11}	0	**48**	0	45	43	40
	10^{-12}	0	**48**	0	45	43	37
$(10, 120)$	10^{-9}	14	**46**	0	44	44	43
	10^{-10}	0	**46**	0	44	44	43
	10^{-11}	0	**46**	0	44	44	43
	10^{-12}	0	**46**	0	44	43	40



(a) $\tilde{f}_1$ (b) $\tilde{f}_2$ (c) $\tilde{f}_3$ (d) $\tilde{f}_4$

(e) $\tilde{f}_5$ (f) $\tilde{f}_1$ (g) $\tilde{f}_2$ (h) $\tilde{f}_3$

(i) $\tilde{f}_4$ (j) $\tilde{f}_5$

Fig. 3.1 Perfect low-rank matrix completion of a rank-10 5000×5000 matrix without any outliers using different smoothing functions in Table 3.1. (a)–(e) comprise the running iteration comparison; (f)–(j) comprise the time comparison.

Next, we ran the same experiment on larger 5000×5000 matrices, with 5% outliers. Fig. 3.3 illustrates the results of these experiments, with $\mu_N = \sigma_N = 0.1$ and $\mu_N = \sigma_N = 1$. In most cases, trust regions still outperformed the other methods in terms of number of iterations, while BFGS performed poorly. Barzilai-Borwein and the conjugate-gradient method were almost as good in terms of time.

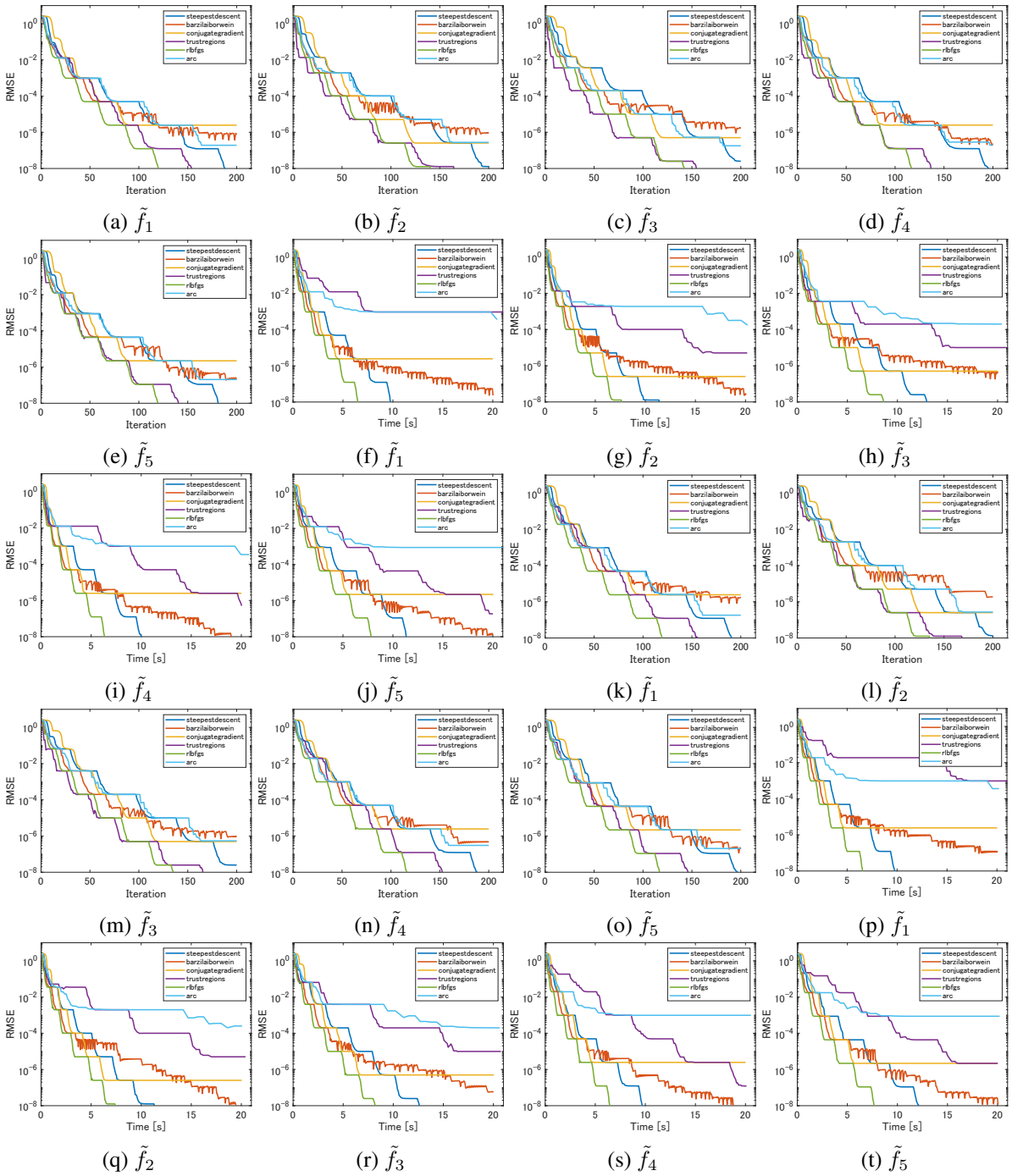
3.8 Note and References

We conclude this chapter by discussing the connections with [43] and [214]. Our work is based on an efficient unification of [43] and [214]. [43] focused on a specific algorithm and did not discuss the underlying generalities, whereas we studied a general framework for Riemannian smoothing. Recall that the "smoothing function" is the core tool of the smoothing method. In addition to what are required by its definition (see Definition 36), it needs to have the following Additional Properties (AP) in order for the algorithms to converge:

(AP1) Approximate from above, i.e., Definition 39. (Needed in Algorithm 1)

(AP2) Gradient sub-consistency, i.e., Definition 37. (Needed in Algorithm 2)

We find that not all smoothing functions satisfy (AP1) and for some functions it is hard to prove whether (AP2) holds. For example, all the functions in Table 3.1 are smoothing functions of $|x|$, but only the first three meet (AP1); the last two do not. In [52], the authors showed that the first one in Table 3.1, $\tilde{f}_1(x, \mu)$,



(a) $\tilde{f}_1$ (b) $\tilde{f}_2$ (c) $\tilde{f}_3$ (d) $\tilde{f}_4$

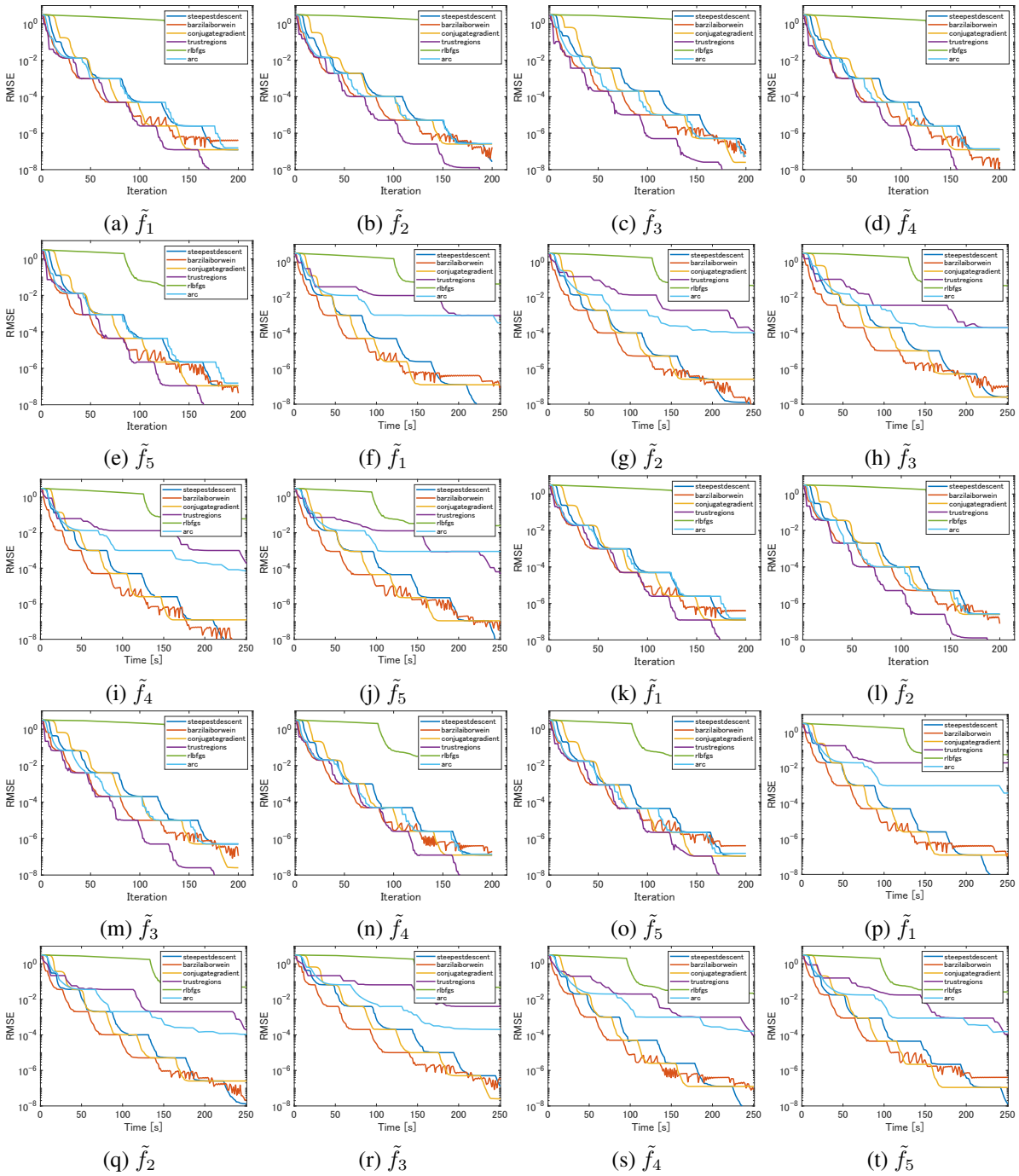
(e) $\tilde{f}_5$ (f) $\tilde{f}_1$ (g) $\tilde{f}_2$ (h) $\tilde{f}_3$

(i) $\tilde{f}_4$ (j) $\tilde{f}_5$ (k) $\tilde{f}_1$ (l) $\tilde{f}_2$

(m) $\tilde{f}_3$ (n) $\tilde{f}_4$ (o) $\tilde{f}_5$ (p) $\tilde{f}_1$

(q) $\tilde{f}_2$ (r) $\tilde{f}_3$ (s) $\tilde{f}_4$ (t) $\tilde{f}_5$

Fig. 3.2 Low-rank matrix completion with outliers for two rank-10 500×500 matrices by using different smoothing functions in Table 3.1. (a)–(j) corresponds to one matrix with outliers created by using $\mu_N = \sigma_N = 0.1$, while (k)–(t) corresponds to the other with outliers created by using $\mu_N = \sigma_N = 1$. (a)–(e) and (k)–(o) comprise the running iteration comparison; (f)–(j) and (p)–(t) comprise the time comparison.



(a) $\tilde{f}_1$ (b) $\tilde{f}_2$ (c) $\tilde{f}_3$ (d) $\tilde{f}_4$

(e) $\tilde{f}_5$ (f) $\tilde{f}_1$ (g) $\tilde{f}_2$ (h) $\tilde{f}_3$

(i) $\tilde{f}_4$ (j) $\tilde{f}_5$ (k) $\tilde{f}_1$ (l) $\tilde{f}_2$

(m) $\tilde{f}_3$ (n) $\tilde{f}_4$ (o) $\tilde{f}_5$ (p) $\tilde{f}_1$

(q) $\tilde{f}_2$ (r) $\tilde{f}_3$ (s) $\tilde{f}_4$ (t) $\tilde{f}_5$

Fig. 3.3 Low-rank matrix completion with outliers for two rank-10 5000×5000 matrices by using different smoothing functions in Table 3.1. (a)–(j) corresponds to one matrix with outliers created by using $\mu_N = \sigma_N = 0.1$, while (k)–(t) corresponds to the other with outliers created by using $\mu_N = \sigma_N = 1$. (a)–(e) and (k)–(o) comprise the running iteration comparison; (f)–(j) and (p)–(t) comprise the time comparison.

has property (AP2). The others remain to be verified, but doing so will not be a trivial exercise. To a certain extent, Algorithm 3 as well as Theorem 11 guarantee a fundamental convergence result even if one has difficulty in showing whether one's smoothing function satisfies (AP2). Therefore, it makes sense to consider Algorithms 3 and 4 together for the sake of the completeness of the general framework.

Algorithm 4 expands on the results of [214]. It allows us to use any standard method of (SRO), not just steepest descent, to solve the smoothed subproblem (3.13). Various standard Riemannian algorithms for (SRO), such as the Riemannian conjugate gradient method [168] (which often performs better than Riemannian steepest descent), the Riemannian Newton method [5, Chapter 6], and the Riemannian trust region method [5, Chapter 7], have extended the concepts and techniques used in Euclidean space to Riemannian manifolds.

As shown by Theorem 12, no matter what kind of sub-algorithm is implemented for (3.13), it does not affect the final convergence as long as the chosen sub-algorithm has property (3.15). On the other hand, we advocate that the sub-algorithm should be viewed as a "Black Box" and the user should not have to care about the code details of the sub-algorithm at all. We can directly use an existing solver, e.g., Manopt [37], which includes the standard Riemannian algorithms mentioned above. Hence, we can choose the most suitable sub-algorithm for the application and quickly implement it with minimal effort.

Chapter 4

Application of RSM: Completely Positive Factorization Problem

Part	Section
Part 1. Preparation work	4.1 Introduction of CP Factorization Problem 4.2 Related Literature 4.3 CP Rank and CP Plus Rank
Part 2. Core proposal	4.4 Reformulation as a Feasibility Problem 4.5 Riemannian Approach to Feasibility Problem
Part 3. Experiments and notes	4.6 Numerical Experiments 4.7 Note and References

In this chapter, we will examine the problem of finding a Completely Positive (CP) factorization of a given completely positive matrix and then treat it as a nonsmooth Riemannian optimization problem. Moreover, we will solve it by using the Riemannian Smoothing Method (RSM) previously introduced in Chapter 3. The numerical experiments clarified that our method can compete with other efficient CP factorization methods, in particular on large-scale matrices.

4.1 Introduction of CP Factorization Problem

The space of $n \times n$ real symmetric matrices $\mathrm{S}(n)$ is endowed with the trace inner product $\langle A, B\rangle := \mathrm{trace}(AB)$.

Definition 40 (Completely Positive Matrix, Copositive Matrix)**.** A matrix $A \in \mathrm{S}(n)$ is called completely positive if for some $r \in \mathbb{N}$ there exists an entrywise nonnegative matrix $B \in \mathbb{R}^{n\times r}$ such that $A = BB^T$, and we call B a CP factorization of A. We define $\mathrm{CP}(n)$ as the set of $n \times n$ completely positive matrices, equivalently characterized as

$$\mathrm{CP}(n) := \{BB^T \in \mathrm{S}(n) : B \text{ is a nonnegative matrix } \} = \mathrm{conv}\{xx^T : x \in \mathbb{R}^n_+\},$$

where "conv" denotes the convex hull. We denote the set of $n \times n$ copositive matrices by

$$\mathrm{COP}(n) := \{A \in \mathrm{S}(n) : x^T Ax \geq 0 \text{ for all } x \in \mathbb{R}^n_+\}.$$

It is known that COP_n and $\mathrm{CP}(n)$ are duals of each other under the trace inner product; moreover, both $\mathrm{CP}(n)$ and $\mathrm{COP}(n)$ are proper convex cones [19, Section 2.2]. For any positive integer n, we have the following inclusion relationship among other important cones in conic optimization:

$$\mathrm{CP}(n) \subset (\mathrm{S}^+(n) \cap \mathrm{N}(n)) \subset \mathrm{S}^+(n) \subset (\mathrm{S}^+(n) + \mathrm{N}(n)) \subset \mathrm{COP}(n),$$

where $\mathrm{S}^+(n)$ is the cone of $n \times n$ symmetric positive semidefinite matrices and $\mathrm{N}(n)$ is the cone of $n \times n$ symmetric nonnegative matrices. See the monograph [19] for a comprehensive description of $\mathrm{CP}(n)$ and $\mathrm{COP}(n)$.

Conic optimization is a sub-field of convex optimization that studies minimization of linear functions over proper cones. Here, if the proper cone is $\mathrm{CP}(n)$ or its dual cone COP_n, we call the conic optimization problem a copositive programming problem. Copositive programming is closely related to many nonconvex, NP-hard quadratic and combinatorial optimizations [68]. For example, consider the so-called standard quadratic optimization problem, $\min\{x^T Mx \mid \mathbf{1}^T x = 1, x \in \mathbb{R}^n_+\}$, where $M \in \mathrm{S}(n)$ is possibly not positive semidefinite and $\mathbf{1}$ is the all-ones vector. Let $\mathbf{E}$ be the all-ones matrix. Bomze et al. [26] showed that the following completely positive reformulation, $\min\{\langle M, X\rangle \mid \langle \mathbf{E}, X\rangle = 1, X \in \mathrm{CP}(n)\}$, is equivalent to the standard quadratic optimization problem. Burer [40] reported a more general result, where any quadratic problem with binary and continuous variables can be rewritten as a linear program over $\mathrm{CP}(n)$. As an application to combinatorial problems, consider the problem of computing the independence number $\alpha(G)$ of a graph G with n nodes. De Klerk and Pasechnik [57] showed that $\alpha(G) = \max\{\langle \mathbf{E}, X\rangle \mid \langle A + I, X\rangle = 1, X \in \mathrm{CP}(n)\}$, where A is the adjacency matrix of G. For surveys on applications of copositive programming, see [23, 27, 41, 67, 68].

The difficulty of the above problems lies entirely in the completely positive conic constraint. Note that because neither $\mathrm{COP}(n)$ nor $\mathrm{CP}(n)$ is self-dual, the primal-dual interior point method for conic optimization does not work as is. Besides this, there are many open problems related to completely positive cones. One is checking membership in $\mathrm{CP}(n)$, which was shown to be NP-hard by [62]. Computing or estimating the cp-rank, as defined later in (4.2), is also an open problem. We refer the reader to [18, 67] for a detailed discussion of those unresolved issues. In this chapter, we focus on the following problem.

Problem 6 (CP Factorization Problem)**.** Finding a CP factorization for a given $A \in \mathrm{CP}(n)$, i.e., the CP factorization problem:

$$\text{find } B \in \mathbb{R}^{n \times r} \text{ s.t. } A = BB^T \text{ and } B \geq 0, \tag{CPfact}$$

(CPfact) seems to be closely related to the membership problem $A \in \mathrm{CP}(n)$. Sometimes, a matrix is shown to be completely positive through duality, or rather, $\langle A, X\rangle \geq 0$ for all $X \in \mathrm{COP}_n$, but in this case, a CP factorization will not necessarily be obtained.

4.2 Related Literature

Various methods of solving CP factorization problems have been studied. In this section, we briefly discuss these related works.

Jarre and Schmallowsky [114] stated a criterion for complete positivity, based on the augmented primal dual method to solve a particular second-order cone problem. Dickinson and Dür [61] dealt with complete positivity of matrices that possess a specific sparsity pattern and proposed a method for finding CP factorizations of these special matrices that can be performed in linear time. Nie [146] formulated the CP factorization problem as an $\mathcal{A}$-truncated K-moment problem, for which the author developed an algorithm that solves a series of semidefinite optimization problems. Sponsel and Dür [183] considered the problem of projecting a matrix onto $\mathrm{CP}(n)$ and $\mathrm{COP}(n)$ by using polyhedral approximations of these cones. With the help of these projections, they devised a method to compute a CP factorization for any matrix in the interior of $\mathrm{CP}(n)$. Bomze [24] showed how to construct a CP factorization of an $n \times n$ matrix based on a given CP factorization of an $(n-1) \times (n-1)$ principal submatrix. Dutour Sikirić et al. [179] developed a simplex-like method for a rational CP factorization that works if the input matrix allows a rational CP factorization.

In 2020, Groetzner and Dür [89] applied the alternating projection method to the CP factorization problem by posing it as an equivalent feasibility problem (see (FeasCP)). Shortly afterwards, Chen et al. [48] reformulated the split feasibility problem as a difference-of-convex optimization problem and solved (FeasCP) as a specific application. In fact, we will solve this equivalent feasibility problem (FeasCP) by other means in this thesis. In 2021, Boţ and Nguyen [33] proposed a projected gradient method with relaxation and inertia parameters for the CP factorization problem, aimed at solving

$$\min_X \{\|A - XX^T\|_{\mathrm{F}}^2 \mid X \in \mathbb{R}_+^{n\times r} \cap \mathscr{B}\}, \tag{4.1}$$

where $\mathscr{B} := \{X \in \mathbb{R}^{n\times r} : \|X\|_{\mathrm{F}} \leq \sqrt{\mathrm{trace}(A)}\}$ is the closed ball centered at 0. The authors argued that its optimal value is zero if and only if $A \in \mathrm{CP}(n)$.

Inspired by the idea of Groetzner and Dür [89], wherein (CPfact) is shown to be equivalent to a feasibility problem called (FeasCP), we treat the problem (FeasCP) as a nonsmooth Riemannian optimization problem and solve it through a general Riemannian smoothing method.

4.3 CP Rank and CP Plus Rank

First, let us recall some basic properties of completely positive matrices. Generally, many CP factorizations of a given A may exist, and they may vary in their numbers of columns. This gives rise to the following definitions: the cp-rank of $A \in \mathrm{S}(n)$, denoted by $\mathrm{cp}(A)$, is defined as

$$\mathrm{cp}(A) := \min\{r \in \mathbb{N} \mid A = BB^T, B \in \mathbb{R}^{n\times r}, B \geq 0\}, \tag{4.2}$$

where $\mathrm{cp}(A) := \infty$ if $A \notin \mathrm{CP}(n)$. Similarly, we can define the cp-plus-rank as

$$\mathrm{cp}^+(A) := \min\{r \in \mathbb{N} \mid A = BB^T, B \in \mathbb{R}^{n\times r}, B > 0\}. \tag{4.3}$$

Immediately, for all $A \in \mathrm{S}(n)$, we have

$$\mathrm{rank}(A) \leq \mathrm{cp}(A) \leq \mathrm{cp}^+(A). \tag{4.4}$$

Every CP factorization B of A is of the same rank as A, since $\mathrm{rank}(XX^T) = \mathrm{rank}(X)$ holds for arbitrary matrix X. The first inequality of (4.4) comes from the fact that for any CP factorization B, $\mathrm{rank}(A) = \mathrm{rank}(B) \leq$ the number of columns of B. The second is trivial by definition. Note that computing or estimating the cp-rank of any given $A \in \mathrm{CP}(n)$ is still an open problem [18]. The following result gives a tight upper bound of the cp-rank for $A \in \mathrm{CP}(n)$ in terms of the order n.

Theorem 13 ([25, Theorem 4.1]). *For all $A \in \mathrm{CP}(n)$, we have*

$$\mathrm{cp}(A) \leq \mathrm{cp}(n) := \begin{cases} n & \text{for } n \in \{2,3,4\} \\ \frac{1}{2}n(n+1) - 4 & \text{for } n \geq 5. \end{cases}$$

The following result is useful for distinguishing completely positive matrices in either the interior or on the boundary of $\mathrm{CP}(n)$.

Theorem 14 ([60, Theorem 3.8]). *We have*

$$\begin{aligned} \mathrm{int}(\mathrm{CP}(n)) &= \{A \in \mathrm{S}(n) : \mathrm{rank}(A) = n, \mathrm{cp}^+(A) < \infty\} \\ &= \{A \in \mathrm{S}(n) : \mathrm{rank}(A) = n, A = BB^T, B \in \mathbb{R}^{n\times r}, B \geq 0, \\ &\quad B_{:,j} > 0 \text{ for at least one column } B_{:,j} \text{ of } B\}. \end{aligned}$$

4.4 Reformulation as a Feasibility Problem

Groetzner and Dür [89] reformulated the CP factorization problem as an equivalent feasibility problem containing an orthogonality constraint.

Given $A \in \mathrm{CP}(n)$, we can easily get another CP factorization $\widehat{B}$ with r' columns for every integer $r' \geq r$, if we also have a CP factorization B with r columns. The simplest way to construct such an $n \times r'$ matrix $\widehat{B}$ is to append $k := r' - r$ zero columns to B, i.e., $\widehat{B} := [B, 0_{n\times k}] \geq 0$. Another way is called column replication, i.e.,

$$\widehat{B} := [B_{:,1}, \ldots, B_{:,n-1}, \underbrace{\frac{1}{\sqrt{m}}B_{:,n}, \ldots, \frac{1}{\sqrt{m}}B_{:,n}}_{m:=r'-n+1 \text{ columns}}], \tag{4.5}$$

where $B_{:,i}$ denotes the i-th column of B. It is easy to see that $\widehat{B}\widehat{B}^T = BB^T = A$. Recall that definition of $\mathrm{cp}(n)$ is given in Theorem 13. As a result, $r \geq \mathrm{cp}(A)$ if and only if A has a CP factorization B with r columns Let us consider the *orthogonal group* of order r:

$$\mathrm{O}(r) := \left\{X \in \mathbb{R}^{r\times r} : X^TX = XX^T = I_r\right\}.$$

Note that many authors have proved that $BB^T = CC^T$ if and only if there exists $X \in \mathrm{O}(r)$ with $BX = C$ for any $B, C \in \mathbb{R}^{n\times r}$ (see, e.g., [41, Lemma 2.1] and [89, Lemma 2.6]). The next proposition puts the previous results together.

Proposition 8. *Let $A \in \mathrm{CP}(n), r \geq \mathrm{cp}(A), A = \bar{B}\bar{B}^T$, where $\bar{B} \in \mathbb{R}^{n\times r}$ may possibly be not nonnegative. Then there exists an orthogonal matrix $X \in \mathrm{O}(r)$ such that $\bar{B}X \geq 0$ and $A = (\bar{B}X)(\bar{B}X)^T$.*

This proposition tells us that one can find an orthogonal matrix X which can turn a "bad" factorization $\bar{B}$ into a "good" factorization $\bar{B}X$. Let $r \geq \mathrm{cp}(A)$ and $\bar{B} \in \mathbb{R}^{n\times r}$ be an arbitrary (possibly not nonnegative) initial factorization $A = \bar{B}\bar{B}^T$. The task of finding a CP factorization of A can then be formulated as the following feasibility problem.

Problem 7 (Reformulation of Problem 6 as a Feasibility Problem [89])**.**

$$\text{find } X \text{ s.t. } \bar{B}X \geq 0 \text{ and } X \in \mathrm{O}(r). \tag{FeasCP}$$

Note that finding an initial matrix $\bar{B}$ is not difficult. Since a completely positive matrix is necessarily positive semidefinite, one can use Cholesky decomposition or spectral decomposition and then extend it to r columns by using (4.5).

Remark 15. The condition $r \geq \mathrm{cp}(A)$ is necessary; otherwise, (FeasCP) has no solution even if $A \in \mathrm{CP}(n)$. Regardless of the exact value of $\mathrm{cp}(A)$ which is often unknown, one can use $r = \mathrm{cp}(n)$ defined in Theorem 13.

4.5 Riemannian Approach to Feasibility Problem

In this section, we will proposal a Riemannian approach to (FeasCP).

For (FeasCP), [89] applied the so-called *alternating projections method* to (FeasCP). Given an initial decomposition $\bar{B}$ of $A \in \mathrm{CP}(n)$, they defined the polyhedral cone, $\mathscr{P} := \{X \in \mathbb{R}^{r\times r} : \bar{B}X \geq 0\}$, and rewrote (FeasCP) as

$$\text{find } X \text{ s.t. } X \in \mathscr{P} \cap \mathrm{O}(r).$$

Let $\mathrm{Proj}_S(x) := \arg\min_{z\in S} \|z - x\|$ denotes the projection of a point x onto some set S. The alternating projections method is as follows: choose a starting point $X_0 \in \mathrm{O}(r)$; then compute $P_0 := \mathrm{Proj}_{\mathscr{P}}(X_0)$ and $X_1 := \mathrm{Proj}_{\mathrm{O}(r)}(P_0)$, and iterate this process. Computing the projection onto $\mathscr{P}$ amounts to solving a Second-Order Cone Problem (SOCP), while computing the projection onto $\mathrm{O}(r)$ amounts to a singular value decomposition. Note that we need to solve an SOCP alternately at every iteration, which is still expensive in practice. A modified version without convergence involves calculating an approximation of $\mathrm{Proj}_{\mathscr{P}}(X_k)$ by using the Moore-Penrose inverse of $\bar{B}$; for details, see [89, Algorithm 2].

Our way is to use the optimization form. Here, we denote by $\max(\cdot)$ (resp. $\min(\cdot)$) the max-function (resp. min-function)) that selects the largest (resp. smallest) entry of a vector or matrix. Notice that $-\min(\cdot) = \max(-(\cdot))$. We associate (FeasCP) with the following optimization problem:

$$\max_{X\in\mathrm{O}(r)} \{\min(\bar{B}X)\}.$$

For consistency of notation, we turn the maximization problem into a minimization problem:

$$\min_{X\in\mathrm{O}(r)} \{\max(-\bar{B}X)\}. \tag{OptCP}$$

The feasible set, orthogonal group $\mathrm{O}(r)$, is known to be compact [96, Lemma 2.1.8]. In accordance with the extreme value theorem [160, Theorem 4.16], (OptCP) attains the global minimum, say t. Summarizing these observations together yields the following proposition.

Proposition 9. *Set $r \geq \mathrm{cp}(A)$, and let $\bar{B} \in \mathbb{R}^{n\times r}$ be an arbitrary initial factorization of A. Then the following statements are equivalent:*

1. $A \in \mathrm{CP}(n)$.
2. *(FeasCP) is feasible.*
3. *In (OptCP), there exists a feasible solution X such that $\max(-\bar{B}X) \leq 0$; or, $\min(\bar{B}X) \geq 0$.*
4. *In (OptCP), the global minimum $t \leq 0$.*

Since the feasible set $\mathrm{O}(r)$ is a Riemannian manifold (refer to Example 3), we treat the CP factorization problem, i.e., (OptCP) as a problem of minimizing a nonsmooth function over a Riemannian manifold, for which variants of subgradient methods [31], proximal gradient methods [56], and the alternating direction method of multipliers (ADMM) [122] have been studied.

4.6 Numerical Experiments

The numerical experiments in Section 4.6 were performed on a computer equipped with an Intel Core i7-10700 at 2.90GHz with 16GB of RAM using Matlab R2022a. Our Algorithm 4 is implemented in the Manopt framework [37] (version 7.0). The number of iterations to solve the smoothed problem (3.13) with the sub-algorithm is recorded in the total number of iterations. The codes is available at https://github.com/GALVINLAI/RieSmooth.

In this section, we describe numerical experiments that we conducted on CP factorization in which we solved (OptCP) using Algorithm 4, where different Riemannian algorithms were employed as sub-algorithms and $\mathrm{lse}(-\bar{B}X, \mu)$ was used as the smoothing function. To be specific, we used three built-in Riemannian solvers of Manopt 7.0 — Steepest Descent (SD), Conjugate Gradient (CG), and Trust Regions (TR), denoted by SM_SD, SM_CG and SM_TR, respectively. We compared our algorithms with the following non-Riemannian numerical algorithms for CP factorization that were mentioned in subsection 4.2. We followed the settings used by the authors in their papers.

- SpFeasDC_ls [48]: A difference-of-convex functions approach for solving the split feasibility problem, it can be applied to (FeasCP). The implementation details regarding the parameters we used are the same as in the numerical experiments reported in [48, Section 6.1].
- RIPG_mod [33]: This is a projected gradient method with relaxation and inertia parameters for solving (4.1). As shown in [33, Section 4.2], RIPG_mod is the best among the many strategies of choosing parameters.
- APM_mod [89]: A modified alternating projection method for CP factorization; it is described in Section 4.5.

We have shown that $\mathrm{lse}(x, \mu)$ is a smoothing function of $\max(x)$ with gradient consistency. The $\mathrm{lse}(\cdot, \mu)$ of the matrix argument can be simply derived from entrywise operations. Then from the properties of compositions of smoothing functions [22, Proposition 1 (3)], we have that $\mathrm{lse}(-\bar{B}X, \mu)$ is a smoothing function of $\max(-\bar{B}X)$ with gradient consistency.

In practice, it is important to avoid numerical overflow and underflow when evaluating $\mathrm{lse}(x, \mu)$. Overflow occurs when any x_i is large and underflow occurs when all x_i are small. To avoid these problems, we can shift each component x_i by $\max(x)$ and use the following formula:

$$\mathrm{lse}(x, \mu) = \mu \log(\sum_{i=1}^{n} \exp((x_i - \max(x))/\mu)) + \max(x),$$

whose validity is easy to show.

The details of the experiments are as follows. If $A \in \mathrm{CP}(n)$ was of full rank, for accuracy reasons, we obtained an initial $\bar{B}$ by using Cholesky decomposition. Otherwise, $\bar{B}$ was obtained by using spectral decomposition. Then we extended $\bar{B}$ to r columns by column replication (4.5). We set $r = \mathrm{cp}(A)$ if $\mathrm{cp}(A)$ was known or r was sufficiently large. We used `RandOrthMat.m` [178] to generate a random starting point X^0 on the basis of the Gram-Schmidt process.

For our three algorithms, we set $\mu_0 = 100, \theta = 0.8$ and used an adaptive rule (3.17) of $\delta_k := \gamma\mu_k$ with $\gamma = 0.5$. Except for RIPG_mod, all the algorithms terminated successfully at k-th iteration , where $\min(\bar{B}X^k) \geq -10^{-15}$ was attained before the maximum number of iterations (5,000) was reached. In addition, SpFeasDC_ls failed when $\bar{L}_k > 10^{10}$. Regarding RIPG_mod, it terminated successfully when $\|A - X_k X_k^T\|_{\mathrm{F}}^2 / \|A\|_{\mathrm{F}}^2 < 10^{-15}$ was attained before at most 10,000 iterations for $n < 100$, and before at most 50,000 iterations in all other cases. In the tables of this section, we report the rounded success rate (Rate) over the total number of trials, although the definitions of "Rate" in the different experiments (described in Sections 4.1-4.4) vary slightly from one experiment to the other. We will describe them later.

4.6.1 Randomly Generated Instances

We examined the case of randomly generated matrices to see how the methods were affected by the order n or r. The instances were generated in the same way as in [89, Section 7.7]. We computed C by setting $C_{ij} := |B_{ij}|$ for all i, j, where B is a random $n \times 2n$ matrix based on the Matlab command `randn`, and we took $A = CC^T$ to be factorized. In Table 4.2, we set $r = 1.5n$ and $r = 3n$ for the values $n \in \{20, 30, 40, 100, 200, 400, 600, 800\}$. For each pair of n and r, we generated 50 instances if $n \leq 100$ and 10 instances otherwise. For each instance, we initialized all the algorithms at the same random starting point X^0 and initial decomposition $\bar{B}$, except for RIPG_mod. Note that each instance A was assigned only one starting point.

Table 4.2 lists the average time in seconds (Time (s)) and the average number of iterations (Iter.) among the successful instances. For our three Riemannian algorithms, Iter. contains the number of iterations of the sub-algorithm. Table 4.2 also lists the rounded success rate (Rate) over the total number (50 or 10) of instances for each pair of n and r. Boldface highlights the two best results in each row.

As shown in Table 4.2, except for APM_mod, each method had a success rate of 1 for all pairs of n and r. Our three algorithms outperformed the other methods on the large-scale matrices with $n \geq 100$. In particular, SM_CG with the conjugate-gradient method gave the best results.

4.6.2 Specifically Structured Instance

Let $\mathbf{1}_n$ denote the all-ones vector in $\mathbb{R}^n$ and consider the matrix [89, Example 7.1],

$$A_n = \begin{bmatrix} 0 & \mathbf{1}_{n-1}^T \\ \mathbf{1}_{n-1} & I_{n-1} \end{bmatrix}^T \begin{bmatrix} 0 & \mathbf{1}_{n-1}^T \\ \mathbf{1}_{n-1} & I_{n-1} \end{bmatrix} \in \mathrm{CP}(n).$$

Theorem 14 shows that $A_n \in \mathrm{int}(\mathrm{CP}(n))$ for every $n \geq 2$. By construction, it is obvious that $\mathrm{cp}(A_n) = n$. We tried to factorize A_n for the values $n \in \{10, 20, 50, 75, 100, 150\}$ in Table 4.3. For each A_n, using $r = \mathrm{cp}(A_n) = n$ and the same initial decomposition $\bar{B}$, we tested all the algorithms on the same 50 randomly generated starting points, except for RIPG_mod. Note that each instance was assigned 50 starting points.

Table 4.3 lists the average time in seconds (Time (s)) and the average number of iterations (Iter.) among the successful starting points. It also lists the rounded success rate (Rate) over the total number (50) of starting points for each n. Boldface highlights the two best results for each n. We can see from Table 4.3 that the success rates of our three algorithms were always 1, whereas the success rates of the other methods decreased as n increased. Likewise, SM_CG with the conjugate-gradient method gave the best results.

4.6.3 Easy Instance on Boundary

Consider the following matrix from [181, Example 2.7]:

$$A = \begin{bmatrix} 41 & 43 & 80 & 56 & 50 \\ 43 & 62 & 89 & 78 & 51 \\ 80 & 89 & 162 & 120 & 93 \\ 56 & 78 & 120 & 104 & 62 \\ 50 & 51 & 93 & 62 & 65 \end{bmatrix}.$$

The sufficient condition from [181, Theorem 2.5] ensures that this matrix is completely positive and $\mathrm{cp}(A) = \mathrm{rank}(A) = 3$. Theorem 14 tells us that $A \in \mathrm{bd}(\mathrm{CP}(5))$, since $\mathrm{rank}(A) \neq 5$.

We found that all the algorithms could easily factorize this matrix. However, our three algorithms returned a CP factorization B whose smallest entry was as large as possible. In fact, they also maximized the smallest entry in the $n \times r$ symmetric factorization of A, since (OptCP) is equivalent to $\max_{A=XX^T, X\in\mathbb{R}^{n\times r}}\{\min(X)\}$. When we did not terminate as soon as $\min(\bar{B}X^k) \geq -10^{-15}$, for example, after 1000 iterations, our algorithms gave the following CP factorization whose the smallest entry is around $2.8573 \gg -10^{-15}$:

$$A = BB^T \text{, where } B \approx \begin{bmatrix} 3.5771 & 4.4766 & \mathbf{2.8573} \\ 2.8574 & 3.0682 & 6.6650 \\ 8.3822 & 7.0001 & 6.5374 \\ 5.7515 & 2.8574 & 7.9219 \\ 2.8574 & 6.7741 & 3.3085 \end{bmatrix}.$$

4.6.4 Hard Instance on Boundary

Next, we examined how well these methods worked on a hard matrix on the boundary of $\mathrm{CP}(n)$. Consider the following matrix on the boundary taken from [69]:

$$A = \begin{bmatrix} 8 & 5 & 1 & 1 & 5 \\ 5 & 8 & 5 & 1 & 1 \\ 1 & 5 & 8 & 5 & 1 \\ 1 & 1 & 5 & 8 & 5 \\ 5 & 1 & 1 & 5 & 8 \end{bmatrix} \in \mathrm{bd}(\mathrm{CP}(5)).$$

Since $A \in \mathrm{bd}(\mathrm{CP}(5))$ and A is of full rank, it follows from Theorem 14 that $\mathrm{cp}^+(A) = \infty$; i.e., there is no strictly positive CP factorization for A. Hence, the global minimum of (OptCP), $t = 0$, is clear. None of the algorithms could decompose this matrix under our tolerance, 10^{-15}, in the stopping criteria. As was done in [89, Example 7.3], we investigated slight perturbations of this matrix. Given

$$MM^T =: C \in \mathrm{int}(\mathrm{CP}(5)) \text{ with } M = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

we factorized $A_\lambda := \lambda A + (1-\lambda)C$ for different values of $\lambda \in [0, 1)$ using $r = 12 > \mathrm{cp}_5 = 11$. Note that $A_\lambda \in \mathrm{int}(\mathrm{CP}(5))$ provided $0 \leq \lambda < 1$ and A_λ approached the boundary as $\lambda \to 1$. We chose the largest $\lambda = 0.9999$. For each A_λ, we tested all of the algorithms on 50 randomly generated starting points and computed the success rate over the total number of starting points.

Table 4.1 shows how the success rate of each algorithm changes as A_λ approaches the boundary. The table sorts the results from left to right according to overall performance. Except for SM_TR, whose success rate was always 1, the success rates of all the other algorithms significantly decreased as λ increased to 0.9999. Surprisingly, the method of SM_CG, which performed well in the previous examples, seemed unable to handle instances close to the boundary.

Table 4.1 Rate rate of CP factorization of A_λ for values of λ from 0.6 to 0.9999.

λ	SM_TR	SM_SD	RIPG_mod	SM_CG	SpFeasDC_ls	APM_mod
0.6	1	1	1	1	1	0.42
0.65	1	1	1	1	1	0.44
0.7	1	1	1	1	1	0.48
0.75	1	1	1	1	1	0.52
0.8	1	1	1	1	0.96	0.46
0.82	1	1	1	1	0.98	0.4
0.84	1	1	1	1	0.86	0.24
0.86	1	1	1	1	0.82	0.1
0.88	1	1	1	1	0.58	0.18
0.9	1	1	1	1	0.48	0.18
0.91	1	1	1	1	0.4	0.14
0.92	1	1	1	1	0.2	0.18
0.93	1	1	0.98	1	0.22	0.22
0.94	1	1	0.98	1	0.1	0.2
0.95	1	1	1	1	0.12	0.32
0.96	1	1	0.96	0.98	0.06	0.34
0.97	1	1	0.86	0.82	0.06	0.14
0.98	1	1	0.76	0.28	0.02	0
0.99	1	0.68	0.42	0	0	0
0.999	1	0	0.14	0	0	0
0.9999	1	0	0	0	0	0

Table 4.2 CP factorization of random completely positive matrices.

$r = 1.5n$	SM_SD			SM_CG			SM_TR			SpFeasDC_ls			RIPG_mod			APM_mod		
n	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
20	1	0.0409	49	1	0.0394	41	1	0.0514	35	1	**0.0027**	24	1	**0.0081**	1229	0.32	0.3502	2318
30	1	0.0549	58	1	0.0477	44	1	0.0690	36	1	**0.0075**	24	1	**0.0231**	1481	0.04	1.0075	2467
40	1	0.0735	65	1	0.0606	46	1	0.0859	37	1	**0.0216**	46	1	**0.0574**	1990	0	-	-
100	1	0.2312	104	1	**0.1520**	56	1	0.4061	45	1	**0.2831**	109	1	0.8169	4912	0	-	-
200	1	**1.0723**	167	1	**0.5485**	69	1	1.9855	53	1	2.2504	212	1	5.2908	9616	0	-	-
400	1	**14.6**	314	1	**4.1453**	86	1	22.1	69	1	36.9	636	1	90.6	17987	0	-	-
600	1	50.6	474	1	**14.7**	105	1	**46.4**	80	1	140.1	882	1	344.7	26146	0	-	-
800	1	133.3	643	1	**30.0**	109	1	**93.5**	89	1	413.3	1225	1	891.1	34022	0	-	-
$r = 3n$	SM_SD			SM_CG			SM_TR			SpFeasDC_ls			RIPG_mod			APM_mod		
n	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
20	1	0.0597	52	1	0.0551	42	1	0.0842	37	1	**0.0057**	15	1	**0.0105**	1062	0.30	0.7267	2198
30	1	0.0793	59	1	0.0673	45	1	0.1161	39	1	**0.0128**	17	1	**0.0336**	1127	0	-	-
40	1	0.1035	67	1	0.0882	48	1	0.1961	41	1	**0.0256**	19	1	**0.0822**	1460	0	-	-
100	1	**0.5632**	103	1	**0.3395**	57	1	1.4128	50	1	0.8115	86	1	1.1909	4753	0	-	-
200	1	**4.5548**	163	1	**2.4116**	68	1	14.3	65	1	8.1517	184	1	9.2248	9402	0	-	-
400	1	**46.5**	296	1	**19.2**	89	1	77.1	80	1	124.3	453	1	156.6	17563	0	-	-
600	1	**209.0**	446	1	**65.7**	99	1	294.5	76	1	981.8	795	1	616.7	25336	0	-	-
800	1	**609.7**	628	1	**160.4**	114	1	650.7	83	1	4027.4	1070	1	1289.4	26820	0	-	-

Table 4.3 CP factorization of a family of specifically structured instances.

Methods	SM_SD			SM_CG			SM_TR			SpFeasDC_ls			RIPG_mod			APM_mod		
$n(r=n)$	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
10	1	0.0399	71	1	0.0313	49	1	0.0424	45	1	**0.0043**	149	1	**0.0074**	2085	0.80	0.0174	616
20	1	0.0486	85	1	0.0408	63	1	0.0637	55	0.98	**0.0139**	201	0.74	**0.0212**	3478	0.90	0.0591	864
50	1	0.2599	295	1	**0.1073**	101	1	**0.2104**	76	0.98	0.3389	770	0	-	-	0.76	0.6948	1416
75	1	**0.3843**	329	1	**0.1923**	135	1	0.4293	93	0.98	1.0706	1186	0	-	-	0.64	1.4809	1510
100	1	**0.7459**	458	1	**0.3289**	168	1	0.9074	108	0.80	1.6653	1083	0	-	-	0.60	2.8150	1690
150	1	**1.8076**	647	1	**0.7837**	241	1	2.6030	145	0.70	3.7652	1170	0	-	-	0.35	9.9930	2959

4.7 Note and References

In this chapter, we examined the problem of finding a CP factorization of a given completely positive matrix and treated it as a Nonsmooth Riemannian Optimization.

Let us we summarize the relation of our approach to the existing CP factorization methods. Groetzner and Dür [89] and Chen et al. [48] proposed different methods to solve (FeasCP). Boţ and Nguyen [33] tried to solve another model (4.1). However, the methods they used do not belong to the Riemannian optimization techniques, but are rather Euclidean ones, since they treated the set $\mathrm{O}(r) := \{X \in \mathbb{R}^{r\times r} : X^T X = I\}$ as a usual constraint in Euclidean space. By comparison, we recognize the existence of manifolds, namely, the Stiefel manifold $\mathcal{M} = \mathrm{O}(r)$, and use optimization techniques specific to them. This change in perspective suggests the possibility of using the rich variety of Riemannian optimization techniques. As the experiments in Section 4.6 show, our Riemannian approach is faster and more reliable than the Euclidean methods.

As in the other numerical methods, there is no guarantee that Algorithm 4 will find a CP factorization for every $A \in \mathcal{CP}^n$. It follows from Proposition 9 that $A \in \mathrm{CP}(n)$ if and only if the global minimum of (OptCP), say t, is such that $t \leq 0$. Since our methods only converge to a stationary point, Algorithm 4 provides us with a local minimizer at best. We are looking forward to finding a global minimizer of (OptCP) in our future work.

Part III

Proposal II - Riemannian Interior Point Methods (RIPM)

Chapter 5

Riemannian Interior Point Methods (RIPM)

Part	Section
Part 1. Preparation work	5.1 Review of Interior Point Methods 5.2 Riemannian Optimality Conditions 5.4 Riemannian Newton Method 5.7 Prototype Algorithm of RIPM
Part 2. Core proposal	5.3 KKT Vector Field 5.5 Covariant Derivative of KKT Vector Field 5.8 Solving Perturbed Newton Equation Efficiently
Part 3. Core analysis	5.6 Implication of Standard Riemannian Assumptions 5.9 Local Convergence 5.10 Summary

Let $\mathcal{M}$ be a connected, complete d-dimensional Riemannian manifold. Starting from this chapter, we consider the following Constrained Riemannian Optimization (CRO),

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0,\ i = 1, 2, \ldots, m, \\ & h_j(x) = 0,\ j = 1, 2, \ldots, l, \\ & x \in \mathcal{M}, \end{aligned} \tag{CRO}$$

where $f\colon \mathcal{M} \to \mathbb{R}$, $h \equiv (h_1, \ldots, h_l)\colon \mathcal{M} \to \mathbb{R}^l$, and $g \equiv (g_1, \ldots, g_m)\colon \mathcal{M} \to \mathbb{R}^m$ are smooth functions. This problem is also called the nonlinear programming problem on a Riemannian manifold. See Section 1.3 for a detailed overview of (CRO), including its research history and various applications.

In this chapter, we try to extend the interior point algorithms from the Euclidean setting, i.e., $\mathcal{M} \equiv \mathbb{R}^d$ in (CRO), to the Riemannian setting. We call this extension the Riemannian Interior Point Method (RIPM). Under meaningful assumptions in the Riemannian setting, we establish the locally convergence. We also show global convergence in next chapter. Let us now briefly review the history of the interior point methods.

5.1 Review of Interior Point Methods

The advent of interior point methods in the 1980s greatly advanced the field of optimization [202, 213, 88]. By the early 1990s, the success of these methods in linear and quadratic programming ignited interest in using them on nonlinear cases [73, 208]. From the 1990s to the first decade of the 21st century, a large number of interior point methods for nonlinear programming emerged. They proved to be as successful as the linear ones [147, Chapter 19].

A subclass known as *primal-dual* interior point methods is the most efficient practical approach. As described in [132], the primal-dual approach to linear programming was introduced in [136]: it was first developed as an algorithm in [121] and eventually became standard for the nonlinear case as well [73, 208]. Since it seems to be an application of the Newton method for solving the KKT conditions, it has been called the *Newton* interior point method in some of the literature.

Recently, Hirai et al. [95] extended the self-concordance-based interior point methods to Riemannian manifolds. They aimed to minimize a *geodesically convex* (i.e., convex on manifolds) objective $f\colon D \to \mathbb{R}$ defined on a geodesically convex subset $D \subset \mathcal{M}$. In contrast, in (CRO) we do not require any convexity. In practice, many convex functions (in the Euclidean sense) are not geodesically convex on some interested manifolds. For example, for any geodesically convex function defined on a connected, compact Riemannian manifold (e.g., Stiefel manifold), it must be constant [34, Corollary 11.10], which is not of interest in the field of optimization. Thus, (CRO) has a wider applicability.

To our knowledge, interior point methods have yet to be considered for (CRO). Thus, we are trying to fill that gap. Our proposal is a generalization of classical local and global convergence theory of interior point methods for nonlinear programming first proposed by El-Bakry et al. [73]. We will build our proposal step-by-step, starting with the optimality conditions of (CRO).

5.2 Riemannian Optimality Conditions

In the field of optimization, the optimality conditions are critical for studying constrained problems in Euclidean setting. These conditions offer first-order/second-order necessary conditions for a solution to be optimal, provided some Constraint Qualifications (CQs) are satisfied. These classical concepts have been generalized to the manifolds case verbatim (see [209, 17, 194] and [128, Subsection 2.2]). In this section, we summarize these results for the sake of completeness.

5.2.1 First-Order Optimality Conditions

The Lagrangian of (CRO) is defined as

$$\mathcal{L}(x, y, z) := f(x) + \sum_{j=1}^{l} y_j h_j(x) + \sum_{i=1}^{m} z_i g_i(x), \tag{5.1}$$

where vectors $y \in \mathbb{R}^l$ and $z \in \mathbb{R}^m$ are called the Lagrange multipliers corresponding to the equality and inequality constraints, respectively.

Fixing multipliers y and z, the (Riemannian) gradient of Lagrangian with respect to the manifold variable $x \in \mathcal{M}$ is denoted as $\operatorname{grad}_x \mathcal{L}(x, y, z)$, given by

$$\operatorname{grad}_x \mathcal{L}(x, y, z) = \operatorname{grad} f(x) + \sum_{j=1}^{l} y_j \operatorname{grad} h_j(x) + \sum_{i=1}^{m} z_i \operatorname{grad} g_i(x). \tag{5.2}$$

Here, $\operatorname{grad} f(x), \{\operatorname{grad} h_j(x)\}_{i=1}^{l}, \{\operatorname{grad} g_i(x)\}_{i=1}^{m}$ are the gradients of f, h_j, and g_i respectively. We have assumed that at least, those functions are differentiable, then their gradients exist. The active set $\mathbb{A}(x)$ at a point $x \in \mathcal{M}$ is defined as

$$\mathbb{A}(x) := \{i : g_i(x) = 0,\ i = 1, 2, \ldots, m\},$$

and contains the indices of the inequality constraints that are "active", i.e., the inequality constraints that are exactly satisfied at x.

Definition 41 (Linear Independence Constraint Qualification (LICQ))**.** We say that the LICQ holds at $x \in \mathcal{M}$ if the tangent vectors in the set

$$\{\operatorname{grad} h_j(x)\}_{j=1}^{l} \cup \{\operatorname{grad} g_i(x)\}_{i \in \mathbb{A}(x)}$$

are linearly independent in tangent space $T_x\mathcal{M}$.

The LICQ serve as a typical Constraint Qualification (CQ) to ensure that optimality conditions hold at an optimal solution $x \in \mathcal{M}$.

Definition 42 (First-Order Necessary Conditions (KKT Conditions))**.** We say that $x \in \mathcal{M}$ satisfies the Karush-Kuhn-Tucker (KKT) conditions of (CRO) if there exist Lagrange multipliers $y \in \mathbb{R}^l$ and $z \in \mathbb{R}^m$ such that the following hold.

$$\text{(Stationarity)}\ \operatorname{grad} f(x) + \sum_{j=1}^{l} y_j \operatorname{grad} h_j(x) + \sum_{i=1}^{m} z_i \operatorname{grad} g_i(x) = 0_x; \tag{5.3}$$

$$\text{(Primal feasibility)}\ g_i(x) \le 0, \quad \forall i = 1, 2, \ldots, m; \tag{5.4}$$

$$h_j(x) = 0, \quad \forall j = 1, 2, \ldots, l; \tag{5.5}$$

$$\text{(Dual feasibility)}\ z_i \ge 0, \quad \forall i = 1, 2, \ldots, m; \tag{5.6}$$

$$\text{(Complementarity)}\ z_i g_i(x) = 0, \quad \forall i = 1, 2, \ldots, m. \tag{5.7}$$

Theorem 15 ([209, Theorem 4.4])**.** *Suppose that $x \in \mathcal{M}$ is a local minimum of (CRO) and that the LICQ holds at x. Then, x satisfies the KKT conditions (5.3)-(5.7).*

Although we assume that LICQ holds under our consideration, some weaker CQs are also available on manifolds. In particularly, [17] generalized the many other CQs for (CRO) and established a chain of implications among these CQs:

$$\text{LICQ} \Longrightarrow \text{MFCQ} \Longrightarrow \text{ACQ} \Longrightarrow \text{GCQ}.$$

Here, the Mangasarian-Fromovitz Constraint Qualification (MFCQ), Abadie's Constraint Qualification (ACQ), and Guignard's Constraint Qualification (GCQ) have been well-understood in the context of Euclidean spaces, as detailed in works like [147, 20]. Furthermore, [17] showed that the weakest one, i.e., GCQ can replace the role of LICQ in Theorem 15, while its claim remains unchanged.

[17] also demonstrated that the stationary condition (5.3), which involves Riemannian gradients (and thus is implicitly determined by a Riemannian metric), can be substituted by (see Definition 19)

$$\mathrm{D}f(x) + \sum_{j=1}^{l} y_j \mathrm{D}h_j(x) + \sum_{i=1}^{m} z_j \mathrm{D}g_j(x) = 0,$$

where those differentials are well-defined without using any Riemannian metric on $\mathcal{M}$, and the right 0 means the zero map from $T_x\mathcal{M}$ to $\mathbb{R}$. This implies that the KKT conditions are intrinsic optimality conditions for (CRO) and independent of Riemannian metric of $\mathcal{M}$.

5.2.2 Second-Order Optimality Conditions

Going one step further, the (Riemannian) Hessian of Lagrangian in (5.1) with respect to $x \in \mathcal{M}$ is denoted as $\mathrm{Hess}_x\, \mathcal{L}(x, y, z)$, i.e., a linear operator on $T_x\mathcal{M}$ such that, for any $\xi \in T_x\mathcal{M}$,

$$\mathrm{Hess}_x\, \mathcal{L}(x, y, z)[\xi] = \mathrm{Hess}\, f(x)[\xi] + \sum_{j=1}^{l} y_j\, \mathrm{Hess}\, h_j(x)[\xi] + \sum_{i=1}^{m} z_i\, \mathrm{Hess}\, g_i(x)[\xi], \tag{5.8}$$

or simply, as a linear combination of linear operators on $T_x\mathcal{M}$:

$$\mathrm{Hess}_x\, \mathcal{L}(x, y, z) = \mathrm{Hess}\, f(x) + \sum_{j=1}^{l} y_j\, \mathrm{Hess}\, h_j(x) + \sum_{i=1}^{m} z_i\, \mathrm{Hess}\, g_i(x). \tag{5.9}$$

Here, $\mathrm{Hess}\, f(x)$, $\{\mathrm{Hess}\, h_j(x)\}_{i=1}^{l}$, $\{\mathrm{Hess}\, g_i(x)\}_{i=1}^{m}$ are Hessians of f, h_j, and g_i respectively. We have assumed that at least, those functions are twice differentiable, then their Hessians exist.

To identify the second-order optimality conditions at a point x with associated Lagrange multipliers y and z, we consider the *critical cone* $\mathcal{C}(x, y, z)$ inside the tangent space $T_x\mathcal{M}$ defined as follows (see [209, Subsection 4.2] and [147, Section 12.4]).

$$\mathcal{C}\,(x, y, z) := \left\{ \xi \in T_x\mathcal{M} \;\middle|\; \begin{array}{l} \langle \xi, \mathrm{grad}\, h_j\,(x)\rangle = 0, \text{ for all } j = 1, 2, \cdots, l, \\ \langle \xi, \mathrm{grad}\, g_i\,(x)\rangle = 0, \text{ for all } i \in \mathbb{A}(x) \text{ with } z_i > 0, \\ \langle \xi, \mathrm{grad}\, g_i\,(x)\rangle \le 0, \text{ for all } i \in \mathbb{A}(x) \text{ with } z_i = 0. \end{array} \right\}. \tag{5.10}$$

Definition 43 (Second-Order Necessary Conditions (SONC))**.** We say that $x \in \mathcal{M}$ satisfies the SONC if it satisfies KKT conditions (5.3)-(5.7) with associated Lagrange multipliers y and z, and

$$\langle \mathrm{Hess}_x\, \mathcal{L}\,(x, y, z)\,\xi, \xi\rangle \ge 0 \text{ for any } \xi \in \mathcal{C}\,(x, y, z)\,.$$

Definition 44 (Second-Order Sufficient Conditions (SOSC))**.** We say that $x \in \mathcal{M}$ satisfies the SOSC if it satisfies KKT conditions (5.3)-(5.7) with associated Lagrange multipliers y and z, and

$$\langle \mathrm{Hess}_x \mathcal{L}(x, y, z)\, \xi, \xi \rangle > 0 \text{ for any nonzero } \xi \in \mathcal{C}(x, y, z).$$

Theorem 16 ([209, Theorem 4.7])**.** *Suppose that $x \in \mathcal{M}$ is a local minimum of (CRO) and that the LICQ holds at x. Then, x satisfies the SONC (see Definition 43).*

Theorem 17 ([209, Theorem 4.8])**.** *Suppose that $x \in \mathcal{M}$ satisfies the SOSC (see Definition 44). Then, it is a strict local minimum of (CRO).*

The second-order optimality conditions will be further discussed in Section 5.6 to guarantee the nonsingularity of covariant derivative of KKT vector field, which is a important result for our Riemannian interior point method. We introduce the concept of KKT vector field in next section.

5.3 KKT Vector Field

In this section, using some more concise notations, we first revisit the KKT conditions (5.3)-(5.7) and then transform those into a vector field defined on a product manifold. Following common usage in the literature about interior-point methods, big letters denote the associated diagonal matrix, e.g.,

$$\begin{aligned} Z &\equiv \mathrm{Diag}(z_1, 2, \ldots, z_n) \text{ for some } z \in \mathbb{R}^n, \\ S &\equiv \mathrm{Diag}(s_1, 2, \ldots, s_n) \text{ for some } s \in \mathbb{R}^n. \end{aligned}$$

For the elements of product manifold, the components in parentheses are sometimes arranged vertically and sometimes horizontally. For example,

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} \equiv (x, y, z) \in \mathcal{M}_1 \times \mathcal{M}_2 \times \mathcal{M}_3$$

for some product manifold $\mathcal{M}_1 \times \mathcal{M}_2 \times \mathcal{M}_3$. Based on above notations and (5.2), the Riemannian KKT conditions (5.3)-(5.7) are now reformulated as

$$\begin{cases} \mathrm{grad}_x \mathcal{L}(x, y, z) = 0_x, \\ h(x) = 0, \\ g(x) \leq 0, \\ Zg(x) = 0, \\ z \geq 0. \end{cases} \tag{5.11}$$

Let $\mathbf{1}$ be the all-ones column vector. With slack variables $s := -g(x)$, furthermore, the above can be written as

$$F(w) := \begin{pmatrix} \operatorname{grad}_x \mathcal{L}(x, y, z) \\ h(x) \\ g(x) + s \\ ZS\mathbf{1} \end{pmatrix} = 0 := \begin{pmatrix} 0_x \\ 0 \\ 0 \\ 0 \end{pmatrix}, \tag{5.12}$$

and $(z, s) \geq 0$, where

$$w := (x, y, z, s) \in \mathcal{N} := \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m. \tag{5.13}$$

Notice that for any $w = (x, y, z, s)$ in $\mathcal{N}$, the first component of $F(w)$ is a tangent vector in $T_x\mathcal{M}$; the Lagrange multipliers y, z and slack variables s, in turn, are treated as they usually are. In fact, we generate a vector field F on the Riemannian product manifold $\mathcal{N}$, i.e.,

$$F \colon \mathcal{N} \to T\mathcal{N} \cong T\mathcal{M} \times T\mathbb{R}^l \times T\mathbb{R}^m \times T\mathbb{R}^m,$$

where $T\mathcal{N}$ denotes the tangent bundle of $\mathcal{N}$, and

$$T_w\mathcal{N} \cong T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$$

under the canonical identification $T_v\mathcal{E} \cong \mathcal{E}$ for any vector space $\mathcal{E}$ and any $v \in \mathcal{E}$. The readers may refer to Section 2.20 for the details of geometry tools on product manifold $\mathcal{N}$.

Definition 45 (KKT Vector Field)**.** Let $\mathcal{N} = \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$. The vector field F on $\mathcal{N}$ defined in (5.12) is called the *KKT vector field* of (CRO).

In summary, the KKT conditions (5.3)-(5.7) for (CRO) can be interpreted as ones for finding a singularity (see Definition 46 in next section) of a vector field on a Riemannian product manifold but with partial nonnegative requirements, namely,

$$\text{Find } w \in \mathcal{N} \text{ such that } F(w) = 0 \text{ and } (z, s) \geq 0. \tag{5.14}$$

Notice that the goal of all the algorithms explored in this thesis is to find points that satisfy the above first-order necessary conditions. Such points, denoted as x^*, or $w^* = (x^*, y^*, z^*, s^*)$, are termed *KKT points*.

Remark 16 (Notes about KKT points)**.** In contrast to convex optimization, where KKT conditions also serve as sufficient conditions for a global minimizer, manifold constraints are generally nonconvex structures. As nonconvex optimization problems present significant challenges, existing research primarily aims to identify KKT points. Although the KKT points are not ensuring to be the optimal minimizer (either locally or globally optimal), a large number of numerical experiments have indicated that the KKT points are an effective criterion to reduce the objective function.

5.4 Riemannian Newton Method

It is well known that in the Euclidean case, the interior point methods are closely related to the Newton method, which is a powerful tool for finding the roots of nonlinear equations:

$$F(p) = 0 \tag{5.15}$$

where $F\colon \mathbb{R}^n \to \mathbb{R}^n$ is a smooth nonlinear function. One of the most effective subclasses of interior point methods, named *primal-dual* interior point methods (also named *Newton* interior point methods in some literature), can be explained as motivated by the use of Newton method to solve a system of nonlinear equations formed by the KKT conditions. We will track this motivation in the next sections and finally present the interior point method on manifolds accordingly.

This section serves as an introduction to the existing research related to generalized Newton method on Riemannian manifolds [77, 76, 74, 32]; it aims to find the singularity of a vector field defined as follows.

Definition 46 (Singularity of a Vector Field)**.** Consider a general manfiold $\mathcal{M}$. Let $F\colon \mathcal{M} \to T\mathcal{M}$ be a smooth vector field. The singularity of a vector field F is a point $p \in \mathcal{M}$ such that

$$F(p) = 0_p \in T_p\mathcal{M}. \tag{Singularity}$$

Remark 17. When $\mathcal{M} = \mathbb{R}^n$, (Singularity) reduces to nonlinear equations (5.15).

Let ∇ be the Riemannian connection on $\mathcal{M}$. Recall the definition of covariant derivative ∇F, which assigns each point $p \in \mathcal{M}$ a linear operator $\nabla F(p)$ from and to $T_p\mathcal{M}$. Then, the (standard) Riemannian Newton method for solving (Singularity) is stated in Algorithm 5. Just as the usual Newton method requires the solution of a linear equation at each iteration, so does Algorithm 5, except that the linear equation is defined on the tangent space.

Algorithm 5: Riemannian Newton Method for (Singularity)

Input: A vector field F on $\mathcal{M}$, an initial point $p_0 \in \mathcal{M}$ and a retraction R on $\mathcal{M}$.

Output: Sequence $\{p_k\} \subset \mathcal{M}$ such that $\{p_k\} \to p^*$ and $F(p^*) = 0_{p^*} \in T_{p^*}\mathcal{M}$.

Set $k \to 0$;

while *stopping criterion not satisfied* **do**

1. Solve the Newton equation (a linear operator equation on tangent space $T_{p_k}\mathcal{M}$):

$$\nabla F(p_k)\xi_k = -F(p_k) \tag{5.16}$$

to obtain $\xi_k \in T_{p_k}\mathcal{M}$;

2. Compute the next point as $p_{k+1} := \mathrm{R}_{p_k}(\xi_k)$;
3. $k \to k + 1$;

end

In particular, when the vector field F is set to the gradient vector field, i.e., $F = \operatorname{grad} f$ for some smooth scalar field f on $\mathcal{M}$, Algorithm 5 becomes the Riemannian Newton method for solving the

unconstrained Riemannian optimization $\min_{x\in\mathcal{M}} f(x)$. Recall that if f is C^1 and x^* is a local minimizer of $\min_{x\in\mathcal{M}} f(x)$, then $\operatorname{grad} f(x^*) = 0$. By Definition 23, then Newton equation (5.16) reduces to

$$\operatorname{Hess} f(p_k)[\xi_k] = -\operatorname{grad} f(p_k)$$

We do not delve into the Riemannian Newton method for this specific case. If interested, please refer to [5, 164, 1, 7].

Remark 18. In this thesis, the term "Riemannian Newton method" refers specifically to Algorithm 5 for solving general singularity problems (Singularity), and not for solving the optimization problems $\min_{x\in\mathcal{M}} f(x)$.

Recent advancements in the convergence of the Riemannian Newton methods can be categorized into local and global convergence. We assume that p^* is a solution of (Singularity). On the topic of local convergence, the following two notable results stand out. [76] demonstrated that if the map $p \mapsto \nabla F(p)$ is locally Lipschitz continuous at p^* and

$$\nabla F(p^*) \text{ is nonsingular,}$$

then the Algorithm 5 achieves local quadratic convergence. [74] revealed that under the more relaxed condition of mere continuity of the map $p \mapsto \nabla F(p)$ at p^*, if $\nabla F(p^*)$ is also nonsingular, then Algorithm 5 achieves local superlinear convergence. Thus, we can observed that the requirement of nonsingularity of the covariant derivative at the solution is essential. On the topic of global convergence, the damped Newton method on manifolds proposed in [32], guarantees global convergence and does so with a superlinear or quadratic rate.

5.5 Covariant Derivative of KKT Vector Field

If we try to apply the Riemannian Newton method to (5.14) (ignoring the non-negative constraints of z and s for the moment), we must first formulate the covariant derivative of KKT vector field F at arbitrary $w \in \mathcal{N}$, which will be given in Lemma 16.

5.5.1 Full Formulation

Recall that we have defined symbol $w = (x, y, z, s)$ as in (5.13), then we can rewrite the Lagrangian $\mathcal{L}(x, y, z)$ in (5.1) as $\mathcal{L}(w)$, and hence, $\operatorname{grad}_x \mathcal{L}(w) \equiv \operatorname{grad}_x \mathcal{L}(x, y, z)$ in (5.2) and $\operatorname{Hess}_x \mathcal{L}(w) \equiv \operatorname{Hess}_x \mathcal{L}(x, y, z)$ in (5.9) for the sake of simplicity. By using those notations, we have the following important result.

Lemma 16 (Covariant Derivative of KKT Vector Field)**.** *Given any $w \in \mathcal{N}$, for the KKT vector field F defined in (5.12), its covariant derivative $\nabla F(w)\colon T_w\mathcal{N} \to T_w\mathcal{N}$ is the linear operator given by*

$$\nabla F(w)\Delta w = \begin{pmatrix} \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \sum_{i=1}^{l} \Delta y_j \operatorname{grad} h_j(x) + \sum_{i=1}^{m} \Delta z_i \operatorname{grad} g_i(x) \\ \langle \operatorname{grad} h_j(x), \Delta x\rangle_x, \textit{ for } j = 1, 2, \ldots, l \\ \langle \operatorname{grad} g_i(x), \Delta x\rangle_x + \Delta s_i, \textit{ for } i = 1, 2, \ldots, m \\ Z\Delta s + S\Delta z \end{pmatrix} \tag{5.17}$$

where $\Delta w = (\Delta x, \Delta y, \Delta z, \Delta s) \in T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m \cong T_w\mathcal{N}$. *A compact formulation of above will be given in Lemma 17 later.*

Proof. For simplicity, we will consider the case in which (CRO) contains only inequality constraints. Then, the Lagrangian $\mathcal{L}(x, z) = f(x) + \sum_{i=1}^m z_i g_i(x)$ and KKT conditions are

$$\begin{cases} \operatorname{grad}_x \mathcal{L}(x, z) = \operatorname{grad} f(x) + \sum_{i=1}^m z_i \operatorname{grad} g_i(x) = 0_x, \\ g_i(x) z_i = 0, \ i = 1, \cdots, m, \end{cases}$$

and $-g(x) \geq 0, z \geq 0$. To avoid a redundant derivation process, we do not introduce the slack variables $s := -g(x)$. Actually, it could be introduced, but there is no essential difference in the proof. Let $G(x)$ be the associated diagonal matrix of vector $g(x) \in \mathbb{R}^m$; then, the KKT vector field (without introducing the slack variables) is $F \colon \mathcal{M} \times \mathbb{R}^m \to T\mathcal{M} \times T\mathbb{R}^m$ given by

$$F(x, z) = \begin{pmatrix} F_1(x, z) \\ F_2(x, z) \end{pmatrix} = \begin{pmatrix} \operatorname{grad}_x \mathcal{L}(x, z) \\ G(x) z \end{pmatrix},$$

where $F_1 \colon \mathcal{M} \times \mathbb{R}^m \to T\mathcal{M}, F_1(x, z) = \operatorname{grad}_x \mathcal{L}(x, z)$, and $F_2 \colon \mathcal{M} \times \mathbb{R}^m \to T\mathbb{R}^m, F_2(x, z) = G(x)z$. We will compute the covariant derivative $\nabla F(x, z)$ of F at $(x, z) \in \mathcal{M} \times \mathbb{R}^m$ step by step in accordance with Lemma 15. Let $\mathbb{R}^m$ be equipped with the canonical Euclidean connection ∇ and $\mathcal{M}$ with a Riemannian connection ∇, and we do not distinguish those connections with superscripts as in Lemma 15, since they should be clear from context.

Let $(u_x, u_z) \in T_x\mathcal{M} \times \mathbb{R}^m$ be the tangent vector at given point $(x, z) \in \mathcal{M} \times \mathbb{R}^m$.

(1) Fix z and consider $F_1(\cdot, z) \colon \mathcal{M} \to T\mathcal{M}$ (a vector field on $\mathcal{M}$). From the $\mathbb{R}$-linearity of the connection ∇ on $\mathcal{M}$ (see Definition 21), we have

$$\nabla_{u_x} F_1(\cdot, z) = \nabla_{u_x} \operatorname{grad}_x \mathcal{L}(x, z) = \operatorname{Hess}_x \mathcal{L}(x, z)[u_x].$$

(2) Fix x and consider $F_1(x, \cdot) \colon \mathbb{R}^m \to T_x\mathcal{M}$ (a map between two vector spaces). By calculus on vector spaces, we have

$$\mathrm{D}F_1(x, \cdot)(z)[u_z] = \lim_{t \to 0} \frac{F_1(x, z + tu_z) - F_1(x, z)}{t} = \sum_{i=1}^m (u_z)_i \operatorname{grad} g_i(x).$$

(3) Fix x and consider $F_2(x, \cdot) \colon \mathbb{R}^m \to T\mathbb{R}^m$ (indeed, it is a map from $\mathbb{R}^m$ to $\mathbb{R}^m$). By the canonical Euclidean connection in $\mathbb{R}^m$ and calculus as usual, we have

$$\nabla_{u_z} F_2(x, \cdot) = \mathrm{D}F_2(x, \cdot)(z)[u_z] = G(x)u_z = [g_1(x)(u_z)_1, \cdots, g_m(x)(u_z)_m]^T.$$

(4) Fix z and consider $F_2(\cdot, z) \colon \mathcal{M} \to T_z\mathbb{R}^m \cong \mathbb{R}^m$ (a map from $\mathcal{M}$ to $\mathbb{R}^m$). Here, let $F_2^i(\cdot, z) = g_i(x)z_i$ be the component function for $i = 1, \ldots, m$. Since $\mathrm{D}F_2^i(\cdot, z)(x)[u_x] = \langle \operatorname{grad}_x F_2^i(\cdot, z), u_x \rangle_x =$

$\langle z_i \operatorname{grad} g_i(x), u_x \rangle_x$, we have

$$\mathrm{D}F_2(\cdot, z)(x)\,[u_x] = \begin{bmatrix} \mathrm{D}F_2^1(\cdot, z)(x)[u_x] \\ \vdots \\ \mathrm{D}F_2^m(\cdot, z)(x)[u_x] \end{bmatrix} = \begin{bmatrix} z_1 \langle \operatorname{grad} g_1(x), u_x \rangle_x \\ \vdots \\ z_m \langle \operatorname{grad} g_m(x), u_x \rangle_x \end{bmatrix}.$$

(5) Finally, by combining the results for steps (1)-(4), $\nabla F(x, z)\,[u_x, u_z]$ is equal to

$$\left(\operatorname{Hess}_x \mathcal{L}(x, z)\,[u_x] + \sum_{i=1}^{m} (u_z)_i \operatorname{grad} g_i(x), \begin{bmatrix} z_1 \langle \operatorname{grad} g_1(x), u_x \rangle_x + g_1(x)\,(u_z)_1 \\ \vdots \\ z_m \langle \operatorname{grad} g_m(x), u_x \rangle_x + g_m(x)\,(u_z)_m \end{bmatrix} \right).$$

Vertically rewriting it yields

$$\nabla F(x, z)\,[u_x, u_z] = \begin{pmatrix} \operatorname{Hess}_x \mathcal{L}(x, z)\,[u_x] + \sum_{i=1}^{m} (u_z)_i \operatorname{grad} g_i(x) \\ z_i \langle \operatorname{grad} g_i(x), u_x \rangle_x + g_i(x)\,(u_z)_i\,, \text{ for } i = 1, 2, \ldots, m \end{pmatrix} \tag{5.18}$$

At first glance, the above result is somewhat unlike (5.17). But the procedures for the four types of functions, namely, steps (1) to (4) described above, are all situations we can encounter when we introduce the slack variables s and the equality constraints $h(x)$. Since there are 4 variables in (5.12), a careful treatment of 16 steps yields (5.17). We omit these trivial processes in this thesis. □

5.5.2 Compact Formulation

To make the formulation of $\nabla F(w)$ in (5.17) look simpler, we introduce the following symbols. For each $x \in \mathcal{M}$, we define two linear operators $\mathcal{H}_x \colon \mathbb{R}^l \to T_x\mathcal{M}$ and $\mathcal{G}_x \colon \mathbb{R}^m \to T_x\mathcal{M}$ by,

$$\mathcal{H}_x v := \sum_{j=1}^{l} v_j \operatorname{grad} h_j(x), \quad \mathcal{G}_x v := \sum_{i=1}^{m} v_i \operatorname{grad} g_i(x), \tag{5.19}$$

for $v \in \mathbb{R}^l (\mathbb{R}^m)$, respectively. We observe that for every $v \in \mathbb{R}^l, \xi \in T_x\mathcal{M}$,

$$\langle \mathcal{H}_x v, \xi \rangle_x = \sum_{j=1}^{l} v_j \langle \operatorname{grad} h_j(x), \xi \rangle_x = v^T \begin{bmatrix} \langle \operatorname{grad} h_1(x), \xi \rangle_x \\ \vdots \\ \langle \operatorname{grad} h_l(x), \xi \rangle_x \end{bmatrix}.$$

Hence, the adjoint of $\mathcal{H}_x$ is given by $\mathcal{H}_x^* \colon T_x\mathcal{M} \to \mathbb{R}^l$,

$$\mathcal{H}_x^* \xi = \left[\langle \operatorname{grad} h_1(x), \xi \rangle_x, \cdots, \langle \operatorname{grad} h_l(x), \xi \rangle_x \right]^T,$$

which is exactly what is in the second line of (5.17). Similarly, $\mathcal{G}_x^* \colon T_x\mathcal{M} \to \mathbb{R}^m$ is given by

$$\mathcal{G}_x^* \xi = \left[\langle \operatorname{grad} g_1(x), \xi \rangle_x, \cdots, \langle \operatorname{grad} g_m(x), \xi \rangle_x \right]^T.$$

By using those symbols above, we obtain a compact form of (5.17) as follows.

Lemma 17 (Compact Form of Covariant Derivative of KKT Vector Field)**.** *Given any $w \in \mathcal{N}$, for the KKT vector field F defined in (5.12), its covariant derivative $\nabla F(w)\colon T_w\mathcal{N} \to T_w\mathcal{N}$ is the linear operator given by*

$$\nabla F(w)\Delta w = \begin{pmatrix} \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \mathcal{H}_x \Delta y + \mathcal{G}_x \Delta z \\ \mathcal{H}_x^* \Delta x \\ \mathcal{G}_x^* \Delta x + \Delta s \\ Z\Delta s + S\Delta z \end{pmatrix}, \tag{5.20}$$

where $\Delta w = (\Delta x, \Delta y, \Delta z, \Delta s) \in T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m \cong T_w\mathcal{N}$. Moreover, its adjoint $\nabla F(w)^*\colon T_w\mathcal{N} \to T_w\mathcal{N}$ is given by

$$\nabla F(w)^*\Delta w = \begin{pmatrix} \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \mathcal{H}_x \Delta y + \mathcal{G}_x \Delta z \\ \mathcal{H}_x^* \Delta x \\ \mathcal{G}_x^* \Delta x + S\Delta s \\ Z\Delta s + \Delta z \end{pmatrix}. \tag{5.21}$$

Let $\mathbf{J}_F(x) \in \mathbb{R}^{q\times p}$ denote the Jacobian matrix of some function $F\colon \mathbb{R}^p \to \mathbb{R}^q$ at x, and $\mathbf{H}_f(x) \in \mathbb{R}^{p\times p}$ denote the Hessian matrix of some $f\colon \mathbb{R}^p \to \mathbb{R}$ at x. If we consider the Euclidean setting $\mathcal{M} = \mathbb{R}^d$ in (CRO), then $\mathcal{H}_x^*, \mathcal{G}_x^*$ are expressed as the Jacobian matrices of h, g at point x, and $\mathcal{H}_x, \mathcal{G}_x$ are their transposes. In this case, operator $\nabla F(w)$ in (5.20) reduces to the matrix multiplication $\Delta w \mapsto \mathbf{J}_F(w)\Delta w$:

$$\mathbf{J}_F(w)\Delta w = \begin{bmatrix} \mathbf{H}_{\mathcal{L}}(w) & \mathbf{J}_h(x)^T & \mathbf{J}_g(x)^T & 0 \\ \mathbf{J}_h(x) & 0 & 0 & 0 \\ \mathbf{J}_g(x) & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta s \end{bmatrix}. \tag{5.22}$$

where $\mathbf{J}_F(w)$ is the $(d + l + 2m) \times (d + l + 2m)$ Jacobian matrix of F at w. Actually, (5.20) reduces to the matrix form of that in Euclidean interior point method (see [147, Equation (19.6), Page 566]).

Remark 19. Moreover, if $\mathcal{M}$ is a Riemannian submanifold of $\mathbb{R}^n$ equipped with the inherited metric $\langle\cdot,\cdot\rangle$, we can express $\mathcal{H}_x$ and $\mathcal{H}_x^*$ (also, $\mathcal{G}_x$ and $\mathcal{G}_x^*$) by only using Euclidean gradients. Let Proj_x be the orthogonal projector form $\mathbb{R}^n$ onto $T_x\mathcal{M} \subset \mathbb{R}^n$, and $\operatorname{egrad} h_j(x)$ be the Euclidean gradients of h_j at x. It follows from the linearity of Proj_x and equality

$$\operatorname{grad} h_j(x) = \operatorname{Proj}_x[\operatorname{egrad} h_j(x)]$$

(see [34, Proposition 3.61]) that

$$\mathcal{H}_x v = \sum_{j=1}^{l} v_j \operatorname{grad} h_j(x) = \sum_{j=1}^{l} v_j \operatorname{Proj}_x[\operatorname{egrad} h_j(x)] = \operatorname{Proj}_x\left(\sum_{j=1}^{l} v_j \operatorname{egrad} h_j(x)\right).$$

On the other hand, since Proj_x is self-adjoint (recall Section 2.1) and $\operatorname{Proj}_x \xi = \xi$ for every $\xi \in T_x\mathcal{M}$, we have

$$\langle \operatorname{grad} h_j(x), \xi\rangle_x = \langle \operatorname{Proj}_x[\operatorname{egrad} h_j(x)], \xi\rangle_x = \langle \operatorname{egrad} h_j(x), \operatorname{Proj}_x \xi\rangle = \langle \operatorname{egrad} h_j(x), \xi\rangle .$$

Thus,

$$\mathcal{H}_x^*\xi = \left[\langle \operatorname{grad} h_1(x), \xi\rangle, \cdots, \langle \operatorname{grad} h_l(x), \xi\rangle\right]^T.$$

It can be applied verbatim to $\mathcal{G}_x$ and $\mathcal{G}_x^*$.

5.6 Implication of Standard Riemannian Assumptions

At the end of Section 5.4, we discussed the conditions on guaranteeing local convergence of Riemannian Newton method. We note that the requirement of nonsingularity for the covariant derivative at the solution point (i.e., $\nabla F(p^*)$) is of primary importance. In fact, under the meaningful optimality conditions previously introduced in Section 5.2 , we can ascertain the nonsingularity of $\nabla F(w^*)$ for our KKT vector field, as demonstrated in Proposition 10. Let $\mathbb{A}(x) := \{i : g_i(x) = 0\}$ denote the active set at $x \in \mathcal{M}$. We collect those meaningful conditions as the following Assumption 1. We call them the standard Riemannian assumptions for (CRO). Note that the x^* and w^* in (A2)-(A4) all refer to those in (A1).

Assumption 1 (Standard Riemannian Assumptions of (CRO))**.**

(A1) (Existence) There exists $w^* = (x^*, y^*, z^*, s^*)$ satisfying the KKT conditions (5.3)-(5.7), or equivalently (5.11). Here, we introduce the slack variables $s^* := -g(x^*)$.

(A2) (Regularity) LICQ holds at x^* (See Definition 41).

(A3) (Strict complementarity) $(z^*)_i > 0$ if $g_i(x^*) = 0$ for all $i = 1, \cdots, m$.

(A4) (Second-order sufficiency) SOSC holds at x^* (See Definition 44). Under the condition (A3), SOSC is equivalent to say that:
$\langle \operatorname{Hess}_x \mathcal{L}(w^*)\xi, \xi\rangle > 0$ for all nonzero $\xi \in T_{x^*}\mathcal{M}$ satisfying $\langle \xi \operatorname{grad} h_j(x^*)\rangle = 0$ for all $j = 1, 2, \cdots, l$, and $\langle \xi \operatorname{grad} g_i(x^*)\rangle = 0$ for all $i \in \mathbb{A}(x^*)$.

Proposition 10. *Let Assumptions (A1)-(A4) hold at some point w^* for (CRO). Then the operator $\nabla F(w^*)$ in (5.17) is nonsingular (i.e, invertible).*

Proof. This proof omits all the asterisks of the variables. Define $\mathbb{E} := \{1, \ldots, l\}$ and $\mathbb{I} := \{1, \ldots, m\}$. Take some $w = (x, y, z, s) \in \mathcal{N}$ satisfying (A1)-(A4), then we have $s_i = -g_i(x)$ and $z_i s_i = 0$ for all $i \in \mathbb{I}$. For short, let $\mathbb{A} := \mathbb{A}(x) \subset \mathbb{I}$.

Suppose that $\nabla F(w)[\Delta w] = 0$ for some $\Delta w = (\Delta x, \Delta y, \Delta z, \Delta s) \in T_w\mathcal{N} \cong T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$. Δy_i denotes the components of the vector Δy, as do Δz_i, Δs_i. To prove its nonsingularity, we will show that $\Delta w = 0$. Expanding the equation $\nabla F(w)[\Delta w] = 0$ gives

$$\begin{cases} 0 = \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \sum_{j\in\mathbb{E}} \Delta y_j \operatorname{grad} h_j(x) + \sum_{i\in\mathbb{I}} \Delta z_i \operatorname{grad} g_i(x), \\ 0 = \langle \operatorname{grad} h_j(x), \Delta x\rangle, \text{ for all } j \in \mathbb{E}, \\ 0 = \langle \operatorname{grad} g_i(x), \Delta x\rangle + \Delta s_i, \text{ for all } i \in \mathbb{I}, \\ 0 = z_i \Delta s_i + s_i \Delta z_i, \text{ for all } i \in \mathbb{I}. \end{cases} \tag{5.23}$$

Strict complementarity (A3) and the last equalities above imply that $\Delta s_i = 0$ for all $i \in \mathbb{A}$ and $\Delta z_i = 0$ for all $i \in \mathbb{I} \setminus \mathbb{A}$. Substituting those values into the system (5.23) reduces it to

$$\begin{cases} 0 = \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \sum\limits_{j\in\mathbb{E}} \Delta y_j \operatorname{grad} h_j(x) + \sum\limits_{i\in\mathbb{A}} \Delta z_i \operatorname{grad} g_i(x), \\ 0 = \langle \operatorname{grad} h_j(x), \Delta x\rangle\,, \text{ for all } j \in \mathbb{E}, \\ 0 = \langle \operatorname{grad} g_i(x), \Delta x\rangle\,, \text{ for all } i \in \mathbb{A}, \end{cases} \tag{5.24}$$

and $\Delta s_i = -\langle \operatorname{grad} g_i(x), \Delta x\rangle$ for all $i \in \mathbb{I} \setminus \mathbb{A}$. It follows from system (5.24) that

$$\begin{aligned} 0 &= \langle \operatorname{Hess}_x \mathcal{L}(w)\Delta x + \sum_{j\in\mathbb{E}} \Delta y_j \operatorname{grad} h_j(x) + \sum_{i\in\mathbb{A}} \Delta z_i \operatorname{grad} g_i(x),\ \Delta x\rangle \\ &= \langle \operatorname{Hess}_x \mathcal{L}(w)\Delta x, \Delta x\rangle + \sum_{j\in\mathbb{E}} \Delta y_j \langle \operatorname{grad} h_j(x), \Delta x\rangle + \sum_{i\in\mathbb{A}} \Delta z_i \langle \operatorname{grad} g_i(x), \Delta x\rangle \\ &= \langle \operatorname{Hess}_x \mathcal{L}(w)\Delta x, \Delta x\rangle\,, \end{aligned}$$

Thus, from second-order sufficiency (A4), Δx must be zero element. And then $\Delta s_i = 0$ for all $i \in \mathbb{I} \setminus \mathbb{A}$. Next, substituting $\Delta x = 0$ into the first equation in (5.24) yields

$$0 = \sum_{j\in\mathbb{E}} \Delta y_j \operatorname{grad} h_j(x) + \sum_{i\in\mathbb{A}} \Delta z_i \operatorname{grad} g_i(x).$$

The Regularity (A2) implies that the coefficients Δy_j for $j \in \mathbb{E}$ and Δz_i for $i \in \mathbb{A}$ must be zero. This completes the proof. □

The result of Proposition 10 again motivates the use of the Newton method for solving (5.14).

5.7 Prototype Algorithm of RIPM

In this section, we will formally propose the prototype of our Riemannian interior point manifold. This prototype algorithm is indeed a local algorithm of RIPM.

Applying the Riemannian Newton method directly to the KKT vector field $F\colon \mathcal{N} \to T\mathcal{N}$ results in the following Newton equation (see (5.16) without iteration count k) at each iteration:

$$\nabla F(w)\Delta w + F(w) = 0. \tag{5.25}$$

As with the usual interior point method in the Euclidean setting, once the iterates reach the boundary of the feasible region, they are forced to stick to it [202, Page 6]. For the iterates to maintain a sufficient distance from the boundary, we introduce a *perturbed* complementary equation for some number $\mu > 0$ and define

$$F_\mu(w) := F(w) - \mu\hat{e}, \text{ and } \hat{e} \equiv \hat{e}(w) := \begin{pmatrix} 0_x \\ 0 \\ 0 \\ \mathbf{1} \end{pmatrix}. \tag{5.26}$$

Here, $\mathbf{1}$ is the all-ones vector whose dimension is clear in context. Notice that the perturbation term $\hat{e}$, indeed, is a special vector field on $\mathcal{N}$, not a constant, because 0_x is essentially dependent on w and/or x.

Definition 47 (Perturbed KKT Vector Field)**.** For some parameter $\mu > 0$, the vector field F_μ defined in (5.26) is called the *perturbed KKT vector field* of (CRO).

Note that the covariant derivative of the perturbed KKT vector field is the same as that of the original KKT vector field. From the linearity of the connection ∇, we have at any point $w \in \mathcal{N}$ and any $\mu > 0$, we have

$$\nabla F_\mu(w) = \nabla F(w) - \mu \nabla \hat{e}(w) = \nabla F(w), \tag{5.27}$$

where the last equity comes from $\nabla \hat{e}(w)[\Delta w] = (0_x, 0, 0, 0)$ for all $\Delta w \in T_w\mathcal{N}$. Applying the Newton method to $F_\mu(w) = 0$ yields the perturbed Newton equation,

$$\nabla F_\mu(w)\Delta w + F_\mu(w) = 0.$$

From (5.26) and (5.27), this equation is equivalent to

$$\nabla F(w)\Delta w + F(w) = \mu \hat{e},$$

which reduces to the ordinary Newton equation (5.25) as $\mu \to 0$. At this point, we can describe a prototype of the Riemannian Interior Point Method (RIPM) in Algorithms 6.

Algorithm 6: Prototype Algorithm of RIPM for (CRO)

Input: A problem of (CRO), an initial point $w_0 = (x_0, y_0, z_0, s_0) \in \mathcal{N}$ with $(z_0, s_0) > 0$ and a retraction R on $\mathcal{M}$.

Output: Sequence $\{x_k\} \subset \mathcal{M}$.

Set $k \to 0$, $\mu_0 > 0$;

while *stopping criterion not satisfied* **do**

1. Solve the perturbed Newton equation (a linear operator equation on tangent space $T_{w_k}\mathcal{N}$):

$$\nabla F(w_k)\Delta w_k = -F(w_k) + \mu_k \hat{e} \tag{5.28}$$

   to obtain $\Delta w_k = (\Delta x_k, \Delta y_k, \Delta z_k, \Delta s_k) \in T_{w_k}\mathcal{N}$;
2. Compute a step size $0 < \alpha_k \leq 1$ to ensure that $(z_{k+1}, s_{k+1}) > 0$;
3. Compute the next point as $w_{k+1} = (x_{k+1}, y_{k+1}, z_{k+1}, s_{k+1}) := \bar{\mathrm{R}}_{w_k}(\alpha_k \Delta w_k)$;
4. Choose $0 < \mu_{k+1} < \mu_k$;
5. $k \to k + 1$;

end

There are many schemes for choosing step size α_k in above to ensure that $(z_{k+1}, s_{k+1}) > 0$. We here present a simple scheme: choose γ_k with $0 < \hat{\gamma} \leq \gamma_k \leq 1$ for some constant $\hat{\gamma}$ and compute the step size,

$$\alpha_k := \min\left\{1, \gamma_k \min_i \left\{-\frac{(s_k)_i}{(\Delta s_k)_i} \mid (\Delta s_k)_i < 0\right\}, \gamma_k \min_i \left\{-\frac{(z_k)_i}{(\Delta z_k)_i} \mid (\Delta z_k)_i < 0\right\}\right\}. \tag{5.29}$$

This scheme is sufficient to guarantee the local superlinear and quadratic convergence of Algorithm 6, as will be proved in Section 5.9.

On the other hand, the next lemma gives a homotopy (or, continuation) derivation of interior point method (see [147, Chapter 19]). Note that Séguin and Kressner [175] developed continuation methods for Riemannian optimization, which are closely related to our Riemannian interior point method.

Lemma 18. *Under Assumptions (A1)-(A4) at some w^*, there exist a sufficiently small $\bar{\mu} > 0$ and a smooth curve $w\colon [0, \bar{\mu}) \to \mathcal{N}$ such that $w(0) = w^*$ and*

$$F_\mu(w(\mu)) = 0, \ \forall \mu \in [0, \bar{\mu})$$

where F is the perturbed KKT vector field as in Definition 47.

Proof. By Proposition 10, we have that $F_0(w^*) = 0$ and $\nabla F_0(w^*)$ is nonsingular. The proof uses the same technique as in [175, Theorem 3.1]. Roughly speaking, it applies the implicit function theorem to the local coordinate representations of the vector field F and its full-rank Jacobian matrix at the solution. □

This smooth curve $\mu \mapsto w(\mu)$ is called the central path, whose endpoint $w(0) = w^*$ is a solution of (CRO). μ is customarily called the barrier parameter because $F_\mu(w(\mu)) = 0$ can be interpreted as the Riemannian KKT conditions of the following barrier problem:

$$\begin{aligned} \min \quad & f(x) - \mu \textstyle\sum_{i=1}^m \log s_i \\ \text{s.t.} \quad & h(x) = 0, \ g(x) + s = 0, \\ & (x, s) \in \mathcal{M} \times \mathbb{R}^m_+. \end{aligned}$$

5.8 Solving Perturbed Newton Equation Efficiently

The challenge of Algorithm 6 is how to solve the Newton equation (5.28) in an efficient manner. In this section, we will do this in two steps: the first step will be to turn the original *full* Newton equation, which is asymmetric and consists of four variables, into an equivalent *condensed* form, which is symmetric and consists of only two variables. In the second step, an iterative method, namely, Krylov subspace method, is used to solve the *operator* equations directly, avoiding the expensive computational effort of converting them into the usual *matrix* equations.

5.8.1 Condensed Form of Perturbed Newton Equation

In this subsection, we will transform original full system into an equivalent condensed system.

Let us consider Algorithm 6 and omit the iteration count k. Given the current point $w \in \mathcal{N}$ with $(z, s) > 0$, for the KKT vector field $F(w)$ in (5.12), we denote its components by F_x, F_y, F_z, F_s in top-to-bottom order, namely,

$$F_x := \operatorname{grad}_x \mathcal{L}(w), F_y := h(x), F_z := g(x) + s, F_s := ZS\mathbf{1}.$$

By using these symbols and the compact form (5.20), the full (perturbed) Newton equation (5.28) defined on $T_w\mathcal{N} \cong T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$ is expanded as:

$$\begin{pmatrix} \mathrm{Hess}_x\, \mathcal{L}(w)\Delta x + \mathcal{H}_x \Delta y + \mathcal{G}_x \Delta z \\ \mathcal{H}_x^* \Delta x \\ \mathcal{G}_x^* \Delta x + \Delta s \\ Z\Delta s + S\Delta z \end{pmatrix} = \begin{pmatrix} -F_x \\ -F_y \\ -F_z \\ -F_s + \mu \mathbf{1} \end{pmatrix}. \tag{5.30}$$

Not only does this equation contain four variables, but there is no symmetry on the left side of the equation, so it would be unwise to solve it just like that.

We suppose that $(z, s) > 0$. (Recall that in our Algorithm 6, we can ensure $(z_k, s_k) > 0$ for every iteration count k.) From the fourth line of (5.30), we can deduce

$$\Delta s = Z^{-1} \left(\mu \mathbf{1} - F_s - S\Delta z\right).$$

Substituting above into the third line of (5.30), the whole system (5.30) reduces to

$$\begin{pmatrix} \mathrm{Hess}_x\, \mathcal{L}(w)\Delta x + \mathcal{H}_x \Delta y + \mathcal{G}_x \Delta z \\ \mathcal{H}_x^* \Delta x \\ \mathcal{G}_x^* \Delta x - Z^{-1} S \Delta z \end{pmatrix} = \begin{pmatrix} -F_x \\ -F_y \\ -Z^{-1}\mu \mathbf{1} - g(x) \end{pmatrix}. \tag{5.31}$$

Again, from the third line of (5.31), we can deduce

$$\Delta z = S^{-1} \left[Z \left(\mathcal{G}_x^* \Delta x + F_z\right) + \mu \mathbf{1} - F_s\right].$$

Substituting this Δz further into the first line of (5.31) and combining it with the second line of (5.31) yields the following *condensed* Newton equation, which is defined on $T_x\mathcal{M} \times \mathbb{R}^l$:

$$\mathcal{T}(\Delta x, \Delta y) := \begin{pmatrix} \mathcal{A}_w \Delta x + \mathcal{H}_x \Delta y \\ \mathcal{H}_x^* \Delta x \end{pmatrix} = \begin{pmatrix} c \\ q \end{pmatrix}. \tag{5.32}$$

where

$$\begin{aligned} \mathcal{A}_w &:= \mathrm{Hess}_x\, \mathcal{L}(w) + \mathcal{G}_x S^{-1} Z \mathcal{G}_x^*, \\ c &:= -F_x - \mathcal{G}_x S^{-1} \left(Z F_z + \mu \mathbf{1} - F_s\right), \\ q &:= -F_y. \end{aligned} \tag{5.33}$$

Here, c and q are constant vectors. If we defined operator

$$\Psi := \mathcal{G}_x S^{-1} Z \mathcal{G}_x^*, \tag{5.34}$$

then $\mathcal{A}_w = \mathrm{Hess}_x\, \mathcal{L}(w) + \Psi$. Note that both Ψ and $\mathrm{Hess}_x\, \mathcal{L}(w)$ are operators from and to $T_x\mathcal{M}$.

From the discussion above, for any $w \in \mathcal{N}$ with $(z, s) > 0$, the operator $\nabla F(w)$ in (5.20) is nonsingular if and only if the newly defined operator $\mathcal{T}$ in (5.32) is nonsingular. Eventually, it is sufficient for us to solve the equation (5.32) containing the only two variables Δx and Δy. In fact, when we

consider the case of only inequality constraints in (CRO), then Δy vanishes and only a linear equation $\mathcal{A}_w[\Delta x] = c$ on $T_x\mathcal{M}$ needs to be solved.

More importantly, the operator $\mathcal{T}$ in the left side of (5.32) is symmetric, or say self-adjoint (although often indefinite). In following lemmas, we show that the operators Ψ and $\mathcal{A}_w$ are self-adjoint; and thus $\mathcal{T}$ is self-adjoint.

Lemma 19. *In condensed form (5.32),*

- *the linear operators $\Psi, \mathcal{A}_w$ is self-adjoint on $T_x\mathcal{M}$ with inner product $\langle\cdot,\cdot\rangle_x$;*
- *the linear operator $\mathcal{T}$ is self-adjoint on the product vector space $T_x\mathcal{M} \times \mathbb{R}^l$ equipped with the inner product $\langle(\xi_x, \xi_y), (\eta_x, \eta_y)\rangle := \langle\xi_x, \eta_x\rangle_x + \xi_y^T\eta_y$.*

Proof. Ψ is self-adjoint because $\Psi^* = \left(\mathcal{G}_x S^{-1} Z \mathcal{G}_x^*\right)^* = \mathcal{G}_x^* Z S^{-1} \mathcal{G}_x = \mathcal{G}_x^* S^{-1} Z \mathcal{G}_x$. Then, we have $\mathcal{A}_w^* = \operatorname{Hess}_x \mathcal{L}(w)^* + \Psi^* = \operatorname{Hess}_x \mathcal{L}(w) + \Psi = \mathcal{A}_w$. Then, to show that $\mathcal{T}$ is self-adjoint, taking $(\xi_x, \xi_y), (\eta_x, \eta_y) \in T_x\mathcal{M} \times \mathbb{R}^l$, we have

$$\begin{aligned}
\left\langle\left(\mathcal{A}_w\xi_x + \mathcal{H}_x\xi_y, \mathcal{H}_x^*\xi_x\right), (\eta_x, \eta_y)\right\rangle =& \langle\mathcal{A}_w\xi_x + \mathcal{H}_x\xi_y, \eta_x\rangle_x + \langle\mathcal{H}_x^*\xi_x, \eta_y\rangle \\
=& \langle\mathcal{A}_w\xi_x, \eta_x\rangle_x + \langle\mathcal{H}_x\xi_y, \eta_x\rangle_x + \langle\mathcal{H}_x^*\xi_x, \eta_y\rangle \\
=& \langle\xi_x, \mathcal{A}_w\eta_x\rangle_x + \langle\xi_y, \mathcal{H}_x^*\eta_x\rangle + \langle\xi_x, \mathcal{H}_x\eta_y\rangle_x \\
\left\langle(\xi_x, \xi_y), \left(\mathcal{A}_w\eta_x + \mathcal{H}_x\eta_y, \mathcal{H}_x^*\eta_x\right)\right\rangle =& \langle\xi_x, \mathcal{A}_w\eta_x + \mathcal{H}_x\eta_y\rangle_x + \langle\xi_y, \mathcal{H}_x^*\eta_x\rangle.
\end{aligned}$$

This completes the proof. □

Remark 20. We can also see that (5.32) is a saddle point problem defined on Hilbert spaces form its special structure. See [15, 159].

The following theorem states two sufficient conditions to ensure that the $\mathcal{T}$ is nonsingular. They are a direct extension of the classical results [28, Theorem 6]. Note that $\mathcal{H}_x$ in (5.19) is injective if and only if the set $\{\operatorname{grad} h_j(x)\}_{j=1}^l$ is linearly independent in $T_x\mathcal{M}$.

Theorem 18. *In condensed form (5.32), the linear operator $\mathcal{T}$ is nonsingular if one of the following conditions hold:*

(a) $\mathcal{A}_w$ and $\mathcal{H}_x^*\mathcal{A}_w^{-1}\mathcal{H}_x$ are nonsingular;

(b) $\mathcal{H}_x$ is injective and $\mathcal{A}_w$ is positive definite on the null space of $\mathcal{H}_x^*$.

Proof. **(a)** Choose an orthonormal basis of $T_x\mathcal{M} \times \mathbb{R}^l$ to generate an isomorphism between the operators and their the matrix representations. If notation "hat" denotes the corresponding matrix, we have

$$\hat{\mathcal{T}} = \begin{bmatrix} \hat{\mathcal{A}}_w & \hat{\mathcal{H}}_x \\ \hat{\mathcal{H}}_x^T & 0 \end{bmatrix}.$$

Since the invertibility, symmetry, and structure are preserved, then matrices $\hat{\mathcal{A}}_w$ and $\hat{\mathcal{H}}_x^T\hat{\mathcal{A}}_w^{-1}\hat{\mathcal{H}}_x$ are invertible. By the properties of Schur complement for block matrix, $\hat{\mathcal{T}}$ is invertible, i.e., $\mathcal{T}$ is nonsingular.

(b) We prove it in a direct way. Let $\mathcal{T}(\Delta x, \Delta y) = (0_x, 0)$ and we show that $(\Delta x, \Delta y) = (0_x, 0)$. Since

$$\mathcal{T}(\Delta x, \Delta y) = \begin{pmatrix} \mathcal{A}_w\Delta x + \mathcal{H}_x\Delta y \\ \mathcal{H}_x^*\Delta x \end{pmatrix} = \begin{pmatrix} 0_x \\ 0 \end{pmatrix},$$

we have $\mathcal{H}_x^*\Delta x = 0$ by the second line, and

$$\begin{aligned} 0 &= \langle (\Delta x, \Delta y), \mathcal{T}(\Delta x, \Delta y)\rangle \\ &= \langle (\Delta x, \Delta y), (\mathcal{A}_w\Delta x + \mathcal{H}_x\Delta y, \mathcal{H}_x^*\Delta x)\rangle \\ &= \langle \Delta x, \mathcal{A}_w\Delta x\rangle_x + \langle \Delta x, \mathcal{H}_x\Delta y\rangle_x + \langle \Delta y, \mathcal{H}_x^*\Delta x\rangle_x \\ \langle \Delta x, \mathcal{A}_w\Delta x\rangle_x &= \langle \Delta x, \mathcal{A}_w\Delta x\rangle_x + 2\langle \mathcal{H}_x^*\Delta x, \Delta y\rangle . \end{aligned}$$

Since Δx is in the null space of $\mathcal{H}_x^*$ and $\mathcal{A}_w$ is positive definite on the null space of H_x^*, then we must have $\Delta x = 0_x$. Turn back to the first line of (5.8.1), we have $\mathcal{H}_x\Delta y = 0_x$. Because $\mathcal{H}_x$ is injective, we have $\Delta y = 0$. □

5.8.2 Krylov Subspace Methods on Tangent Space

Next, how to solve (5.32) efficiently becomes critical. For simplicity, we consider the case of only inequality constraints in (CRO), then we will only solve

$$\mathcal{A}_w[\Delta x] = c \tag{5.35}$$

with a self-adjoint operator $\mathcal{A}_w\colon T_x\mathcal{M} \to T_x\mathcal{M}$. Let $d := \dim T_x\mathcal{M}$. Unfortunately, in most cases of practical applications, Riemannian situation leaves us with no explicit matrix form available for $\mathcal{A}_w$. This means that we can only access $\mathcal{A}$ (subscript w omitted) by inputting a vector v to return $\mathcal{A}v$. A general approach is to first find the matrix representation $\hat{\mathcal{A}}$ for $\mathcal{A}$ under some basis of $T_x\mathcal{M}$. In detail, the full process of this approach is described in Algorithm 7.

Algorithm 7: General Matrix Representation Method for (5.35)

Input: Sysmetric invertible linear operator $\mathcal{A}\colon T_x\mathcal{M} \to T_x\mathcal{M}$, nonzero $c \in T_x\mathcal{M}$.

Output: Solution Δx of $\mathcal{A}\Delta x = c$.

(Step 1) Obtain d random independent vectors on $T_x\mathcal{M}$ (it often needs d orthogonal projection operations onto subspace $T_x\mathcal{M}$ when $\mathcal{M}$ is a Riemannian submanifold, e.g., `M.randvec` in package Manopt [37]);

(Step 2) Obtain an orthonormal basis $\{u_i\}_{i=1}^d$ of $T_x\mathcal{M}$ by the modified Gram-Schmidt algorithm;

(Step 3) Compute $(\hat{\mathcal{A}})_{ij} := \langle \mathcal{A}u_j, u_i\rangle_x$ for $1 \le i \le j \le d$ due to symmetry, then we obtain the matrix representation $\hat{\mathcal{A}} \in \mathbb{R}^{d\times d}$;

(Step 4) Compute $(\hat{c})_i := \langle c, u_i\rangle_x$ for $1 \le i \le d$, then we obtain the vector representation $\hat{c} \in \mathbb{R}^d$;

(Step 5) Use an arbitrary linear solver to get the solution $\Delta\hat{x} \in \mathbb{R}^d$ of matrix equation $\hat{\mathcal{A}}\Delta\hat{x} = \hat{c}$;

(Step 6) Recovery the tangent vector $\Delta x \in T_x\mathcal{M}$ by $\Delta x = \sum_{i=1}^d (\Delta\hat{x})_i u_i$;

Recall that in Algorithm 6, at each iteration, x is updated and thus the tangent space $T_x\mathcal{M}$ changes (so does operator $\mathcal{A}$). Thus the six Steps in Algorithm 6 need to be done all over again. Obviously, this approach is so expensive that it is not feasible in practice.

An ideal approach is to use an iterative method, such as a Krylov subspace method (e.g., conjugate gradients method [34, Chapter 6.3]), on $T_x\mathcal{M}$ directly. Such a method does not explicitly require a coefficient matrix, and instead needs only a matrix-vector product. In general, it only needs to call an

abstract linear operator $v \mapsto \mathcal{A}v$. Since $\mathcal{A}$ in (20) is self-adjoint but indefinite, for solving operator equation (5.35), we will use the Conjugate Residual (CR) method (see [161, ALGORITHM 6.20]) as stated in Algorithm 8.

Algorithm 8: Conjugate Residual (CR) Method on Tangent Spaces for (5.35)

Input: Sysmetric invertible linear operator $\mathcal{A}\colon T_x\mathcal{M} \to T_x\mathcal{M}$, nonzero $c \in T_x\mathcal{M}$ and an initial point $v_0 \in T_x\mathcal{M}$.

Output: Sequence $\{v_n\} \subset T_x\mathcal{M}$ such that $\{v_n\} \to v^*$ and $\mathcal{A}v^* = c$.

Set $n \to 0$, $r_0 := c - Av_0, p_0 := r_0$ and compute $\mathcal{A}r_0, \mathcal{A}p_0$;

while *stopping criterion not satisfied* **do**

Update number $\alpha_n := \langle r_n, \mathcal{A}r_n\rangle_x / \langle \mathcal{A}p_n, \mathcal{A}p_n\rangle_x$; `// Update step size`

$v_{n+1} := v_n + \alpha_n p_n$; `// Update iterate point`

$r_{n+1} := r_n - \alpha_n \mathcal{A}p_n$; `// Update Residual`

Compute $\mathcal{A}r_{n+1}$; `// This is the only call to` $\mathcal{A}$ `in while loop`

Update number $\beta_n := \langle r_{n+1}, \mathcal{A}r_{n+1}\rangle_x / \langle r_n, \mathcal{A}r_n\rangle_x$;

$p_{n+1} := r_{n+1} + \beta_n p_n$; `// Update conjugate direction`

$\mathcal{A}p_{n+1} := \mathcal{A}r_{n+1} + \beta_n \mathcal{A}p_n$; `// No need to call` $\mathcal{A}$ `here`

$n \to n + 1$;

end

A significant feature is that the iterates v_n, conjugate directions p_n, and residual vectors $r_n := \mathcal{A}v_n - c$ are all contained in $T_x\mathcal{M}$. Usually, the initial point v_0 is the zero element of $T_x\mathcal{M}$; the iteration terminates when the relative residual $\|r_n\|/\|c\| \leq \varepsilon$ for some threshold $\varepsilon > 0$, or some maximum number of iterates is reached.

The discussion of the above two approaches can be naturally extended to the case containing equality constraints, where we consider $\mathcal{T}$ with the product space $T_x\mathcal{M} \times \mathbb{R}^l$ instead of $\mathcal{A}$ with $T_x\mathcal{M}$.

5.9 Local Convergence

In this section, we will prove the local superlinear/quadratic convergence of prototype Algorithm 6. Along the lines of the local convergent analysis in Euclidean setting (see [73, Section 5]), we begin with a perturbed damped Riemannian Newton method and then derive its local convergence theory in Subsection 5.9.1. Then, the convergence of Algorithm 6 results from an application of the perturbed damped Riemannian Newton method and will be described in Subsection 5.9.2 later. Here, for any two nonnegative infinite sequences of reals $\{u_k\}$ and $\{v_k\}$, we write $u_k = O(v_k)$ if there is a constant $M > 0$ such that $u_k \leq Mv_k$ for all sufficiently large k; and we write $u_k = o(v_k)$ if $v_k > 0$ and the sequence of ratios $\{u_k/v_k\}$ approaches zero.

5.9.1 Perturbed Damped Riemannian Newton Method

Let us digress from the optimization problem (CRO) for a moment and refocus on the problem of finding the singularity of a vector field (see Section 5.4) as follows:

$$F(p) = 0_p \in T_p\mathcal{M}, \tag{Singularity}$$

where F is a C^1 vector field on $\mathcal{M}$. We will rely on an application of so-called *perturbed damped* Riemannian Newton method for solving (Singularity), which can be stated as Algorithm 9.

Algorithm 9: Perturbed Damped Riemannian Newton Method for (Singularity)

Input: A vector field F on $\mathcal{M}$, an initial point $p_0 \in \mathcal{M}$ and a retraction R on $\mathcal{M}$.

Output: Sequence $\{p_k\} \subset \mathcal{M}$ such that $\{p_k\} \to p^*$ and $F(p^*) = 0_{p^*} \in T_{p^*}\mathcal{M}$.

Set $k \to 0, \mu_0 > 0$;

while *Stopping condition is not met* **do**

1. Solve the perturbed Newton equation (a linear operator equation on tangent space $T_{p_k}\mathcal{M}$):

$$\nabla F(p_k)\xi_k = -F(p_k) + \mu_k \hat{e} \tag{5.36}$$

   to obtain $\xi_k \in T_{p_k}\mathcal{M}$;
2. Choose a (damped) step size $0 < \alpha_k \leq 1$;
3. Compute the next point as $p_{k+1} := \mathrm{R}_{p_k}(\alpha_k \xi_k)$;
4. Choose $0 < \mu_{k+1} < \mu_k$;
5. $k \to k + 1$;

end

In contrast to the (standard) Riemannian Newton method described in Algorithm 5, the term "perturbed" means that we solve a Newton equation (5.36) with a perturbed term $\mu_k \hat{e}$ (for $\hat{e}$ see (5.26)), while "damped" means using α_k instead of unit steps. It is well known that Algorithm 5 are locally superlinearly [74] and quadratically [77] convergent under the following Standard Riemannian Newton assumptions:

Assumption 2 (Standard Riemannian Newton Assumptions of (Singularity))**.**

(B1) There exists $p^* \in \mathcal{M}$ such that $F(p^*) = 0_{p^*}$.

(B2) The covariant derivative $\nabla F(p^*)$ is nonsingular.

(B3) The map $p \mapsto \nabla F(p)$ is locally Lipschitz continuous at p^*.

As the following Proposition 11 shows, Algorithm 9 also has the same convergence properties as Algorithm 5 if we control μ_k and α_k according to the two schemes that Proposition 11 gives. We can see that either scheme will have $\mu_k \to 0$ and $\alpha_k \to 1$, which makes Algorithm 9 eventually reduce to Algorithm 5 when k is sufficiently large.

Proposition 11 (Local Convergence of Algorithm 9)**.** *Consider the perturbed damped Riemannian Newton method described in Algorithm 9 for singularity problem (Singularity). Let the standard assumptions (B1)-(B3) hold at solution point p^*. If we choose the parameters μ_k and α_k below (i) or (ii), then there exists a constant $\delta > 0$ such that for any initial point p_0 with $d(p_0, p^*) < \delta$, the sequence $\{p_k\}$ in Algorithm 9 is well-defined (i.e., for each k, equation (5.36) has a solution). Furthermore,*

(i) if we choose $\mu_k = o(\|F(p_k)\|)$ *and* $\alpha_k \to 1$*, then* $\{p_k\}$ *converges to* p^* *superlinearly;*

(ii) if we choose $\mu_k = O(\|F(p_k)\|^2)$ *and* $1 - \alpha_k = O(\|F(p_k)\|)$*, then* $\{p_k\}$ *converges to* p^* *quadratically.*

Proof. By Lemma 14, we can let p_k be sufficiently close to p^* such that $\nabla F(p_k)$ is nonsingular, and $\left\|\nabla F(p_k)^{-1}\right\| \leq \Xi$. Then, the next iterate,

$$p_{k+1} := R_{p_k}[\alpha_k \nabla F(p_k)^{-1}(-F(p_k) + \mu_k \hat{e})],$$

is well-defined in Algorithm 9. It follows from $p^* = R_{p_k}(\eta)$ with $\eta := \mathrm{R}_{p_k}^{-1} p^*$ and Lemma 6 that

$$\begin{aligned} d(p_{k+1}, p^*) &\leq a_1 \|\alpha_k \nabla F(p_k)^{-1}(-F(p_k) + \mu_k \hat{e}) - \eta\| \\ &= a_1 \|\eta + \alpha_k \nabla F(p_k)^{-1}(F(p_k) - \mu_k \hat{e})\|. \end{aligned} \tag{5.37}$$

Let $r_k := \eta + \alpha_k \nabla F(p_k)^{-1}(F(p_k) - \mu_k \hat{e})$. Algebraic manipulations show that

$$\begin{aligned} r_k =& (1 - \alpha_k)\eta + \alpha_k \nabla F(p_k)^{-1} \nabla F(p_k)\eta + \alpha_k \nabla F(p_k)^{-1}(F(p_k) - \mu_k \hat{e}) \\ =& (1 - \alpha_k)\eta + \alpha_k \nabla F(p_k)^{-1}[\nabla F(p_k)\eta + F(p_k) - \mu_k \hat{e}] \\ =& (1 - \alpha_k)\eta + \alpha_k \nabla F(p_k)^{-1}[\nabla F(p_k)\eta + F(p_k) - \mathrm{P}_{\gamma}^{0 \to 1} F(p^*) - \mu_k \hat{e}], \end{aligned}$$

where P_γ is a parallel transport along the curve γ given by $\gamma(t) = \mathrm{R}_{p_k}(t\eta)$.

Thus, using $\|\eta\| \leq \frac{1}{a_0} d(p_k, p^*)$ from (iii) of Lemma 6 and the first estimation of Lemma 11, we have

$$\begin{aligned} \|r_k\| &\leq (1 - \alpha_k)\|\eta\| + \alpha_k \|\nabla F(p_k)^{-1}\| \|\mathrm{P}_{\gamma}^{0 \to 1} F(p^*) - F(p_k) - \nabla F(p_k)\eta\| + \alpha_k \|\nabla F(p_k)^{-1}\| \|\hat{e}\| \mu_k \\ &\leq \frac{1}{a_0}(1 - \alpha_k) d(p_k, p^*) + \alpha_k \|\nabla F(p_k)^{-1}\| c_2 d^2(p_k, p^*) + \alpha_k \|\nabla F(p_k)^{-1}\| \|\hat{e}\| \mu_k \\ &\leq \frac{1}{a_0}(1 - \alpha_k) d(p_k, p^*) + \Xi c_2 d^2(p_k, p^*) + \Xi \|\hat{e}\| \mu_k. \text{ (by Lemma 14 and } 0 < \alpha_k \leq 1) \end{aligned}$$

Therefore, by combining the above with (5.37), we conclude that

$$d(p_{k+1}, p^*) \leq \kappa_1 (1 - \alpha_k) d(p_k, p^*) + \kappa_2 d^2(p_k, p^*) + \kappa_3 \mu_k \tag{5.38}$$

for some positive constants $\kappa_1, \kappa_2, \kappa_3$. On the other hand, by Lemma 7, we have

$$\|F(p_k)\| = O(d(p_k, p^*)). \tag{5.39}$$

In what follows, we prove assertions (i) and (ii).

(i) Suppose that $\alpha_k \to 1$ and $\mu_k = o(\|F(p_k)\|)$, which together imply $\mu_k = o(d(p_k, p^*))$. By (5.38), we have

$$\frac{\mathrm{d}(p_{k+1}, p^*)}{d(p_k, p^*)} \leq \kappa_1 (1 - \alpha_k) + \kappa_2 d(p_k, p^*) + \kappa_3 \frac{\mu_k}{d(p_k, p^*)}, \tag{5.40}$$

and we can take δ sufficiently small and k sufficiently large, if necessary, to conclude that

$$d(p_{k+1}, p^*) < \frac{1}{2} d(p_k, p^*) < \delta.$$

Thus, $p_{k+1} \in B_\delta(p^*)$. By mathematical induction, it is easy to show that the sequence $\{p_k\}$ is well-defined and converges to p^*. Taking the limit of both sides of (5.40) proves superlinear convergence.

(ii) Again, we start from (5.38):

$$d(p_{k+1}, p^*) = (1 - \alpha_k)O(d(p_k, p^*)) + O(d^2(p_k, p^*)) + O(\mu_k). \tag{5.41}$$

Suppose that $1 - \alpha_k = O(\|F(p_k)\|)$ and $\mu_k = O(\|F(p_k)\|^2)$. Using (5.39), the above equality reduces to

$$d(p_{k+1}, p^*) = O(d^2(p_k, p^*)).$$

This implies that there exists a positive constant ν such that $d(p_{k+1}, p^*) \leq \nu d^2(p_k, p^*)$, and hence,

$$d(p_{k+1}, p^*) \leq \nu d^2(p_k, p^*) \leq \nu\delta^2 < \delta,$$

if δ is sufficiently small. Again, by mathematical induction, it is easy to show that the sequence $\{p_k\}$ converges to p^* quadratically. □

5.9.2 Superlinear and Quadratic Convergence Theorem

Now, let us establish local convergence of RIPM in a way that almost replicates the results of perturbed damped Newton method. In particular, we will consider Algorithm 6 with step size rule (5.29). We first need the next lemma, which shows the relationship between the parameter γ_k and step size α_k in the rule (5.29).

Lemma 20. *Consider the Algorithm 6 for problem (CRO). Let (A1) and (A3) hold at some* $w^* = (x^*, y^*, z^*, s^*)$. *Suppose that the step size* α_k *is chosen as in (5.29). Define a constant,*

$$\Pi := 2\max\left\{\max_i\left\{\frac{1}{(s^*)_i} \mid (s^*)_i > 0\right\}, \max_i\left\{\frac{1}{(z^*)_i} \mid (z^*)_i > 0\right\}\right\}.$$

For $\gamma_k \in (0, 1)$, *if*

$$\Pi \|\Delta w_k\| \leq \gamma_k, \tag{5.42}$$

then

$$0 \leq 1 - \alpha_k \leq (1 - \gamma_k) + \Pi \|\Delta w_k\|. \tag{5.43}$$

Proof. Notice that the fourth line of (5.30) yields

$$S_k^{-1}\Delta s_k + Z_k^{-1}\Delta z_k = \mu_k(S_k Z_k)^{-1}\mathbf{1} - \mathbf{1},$$

which is exactly the same as in the usual interior point method in the Euclidean setting. We don not need to pay attention to any concept of manifolds. Thus, the proof entails directly applying [208, Lemma 3 and 4] for the Euclidean case to the Riemannian case. □

Remark 21. Rule (5.29) uses a single step size for the all variables. Another popular rule, also mentioned in [208, Equation (3.15)], uses different step sizes as follows: let $w_{k+1} = (\mathrm{R}_{x_k}(\alpha_{x_k}\Delta x_k), y_k +$

$\alpha_{y_k}\Delta y_k, s_k + \alpha_{s_k}\Delta s_k, z_k + \alpha_{z_k}\Delta z_k)$, where

$$\alpha_{s_k} = \min\left\{1, \gamma_k \min_i \left\{-\frac{(s_k)_i}{(\Delta s_k)_i} \mid (\Delta s_k)_i < 0\right\}\right\},$$
$$\alpha_{z_k} = \min\left\{1, \gamma_k \min_i \left\{-\frac{(z_k)_i}{(\Delta z_k)_i} \mid (\Delta z_k)_i < 0\right\}\right\},$$

and $\alpha_{x_k} = \alpha_{y_k}$ are equal to one or α_{s_k} or α_{z_k}. For this step size rule, we can obtain a similar result as in Lemma 20. Refer to [208, Lemma 5] for details.

Now, let us establish the local convergence of our Algorithm 6 in a way that replicates Proposition 11 except for taking account of parameter γ_k.

Theorem 19 (Local Convergence of Prototype Algorithm 6)**.** *Consider the prototype Algorithm 6 for solving problem (CRO). Let (A1)-(A4) hold at some w^*. If we choose the parameters μ_k, γ_k as follows; then there exists a constant $\delta > 0$ such that, for all $w_0 \in \mathcal{N}$ with $d(w_0, w^*) < \delta$, the sequence $\{w_k\}$ is well defined. Furthermore,*

(i) if we choose $\mu_k = o(\|F(w_k)\|)$ and $\gamma_k \to 1$, then $w_k \to w^*$ superlinearly;

(ii) if we choose $\mu_k = O(\|F(w_k)\|^2)$ and $1 - \gamma_k = O(\|F(w_k)\|)$, then $w_k \to w^*$ quadratically.

Proof. We only prove (ii) because (i) can be proven in the same way. Suppose that $d(w_k, w^*) < \delta$ for sufficiently small δ. Since F satisfies assumptions (B1)-(B3), form the proof of Proposition 11 and equation (5.41), we also have

$$d(w_{k+1}, w^*) = (1 - \alpha_k)O(d(w_k, w^*)) + O(d^2(w_k, w^*)) + O(\mu_k).$$

Since $\mu_k = O(\|F(w_k)\|^2)$, and $\|F(w_k)\| = O(d(w_k, w^*))$ by equation (5.39), we obtain

$$\mu_k = O(d^2(w_k, w^*)). \tag{5.44}$$

Thus, we have

$$\begin{aligned}
\|\Delta w_k\| &= \left\|\nabla F(w_k)^{-1}(-F(w_k) + \mu_k \hat{e})\right\| \\
&\le \Xi(\|F(w_k)\| + \mu_k\|\hat{e}\|) \text{ (by Lemma 14)} \\
&\le O(F(w_k)) + O(\mu_k) \\
&= O(d(w_k, w^*)) + O(d^2(w_k, w^*)) = O(d(w_k, w^*)) \text{ (by equation (5.44)).}
\end{aligned}$$

Since δ is sufficiently small, from equation (5.44) and the above inequalities, the conditions of Lemmas 20 are satisfied. Hence, we have

$$0 \le 1 - \alpha_k \le (1 - \gamma_k) + \Pi\,\|\Delta w_k\| = (1 - \gamma_k) + O(d(w_k, w^*)), \tag{5.45}$$

and

$$
\begin{aligned}
&d(w_{k+1}, w^*) = (1-\alpha_k)O(d(w_k, w^*)) + O(d^2(w_k, w^*)) + O(\mu_k) \\
&\leq [(1-\gamma_k) + O(d(w_k, w^*)]O(d(w_k, w^*)) + O(d^2(w_k, w^*)) + O(d^2(w_k, w^*)) \text{ (by (5.44), (5.45))} \\
&= O(d^2(w_k, w^*)).
\end{aligned}
$$

This completes the proof. □

5.10 Summary

This chapter introduces the Riemannian Interior Point Methods (RIPM) applied to a connected, complete Riemannian manifold for solving Constrained Riemannian Optimization (CRO). We extend the classical primal-dual interior point method from the Euclidean setting to the Riemannian one, establishing local convergence under certain assumptions.

Section 5.2 includes an overview of Riemannian optimality conditions, adapting classical concepts of first-order/second-order necessary conditions and Constraint Qualifications (CQs) to manifold settings. Section 5.3 reinterprets the KKT conditions of (CRO) as a special vector field (called KKT vector field) on a Riemannian product manifold. In Section 5.4, the Riemannian Newton Method is discussed, along with the covariant derivative of the KKT vector field given in Section 5.5, essential for applying the interior point method to the (CRO). We provides both full and compact formulations for this covariant derivative. Section 5.6 explores the implications of standard Riemannian assumptions, particularly the importance of the nonsingularity of the covariant derivative at the solution point for local convergence. In Section 5.7, a prototype algorithm of RIPM is proposed, focusing on local algorithms. The challenge of efficiently solving the perturbed Newton equation is addressed in Section 5.8, proposing a two-step approach that simplifies the equation and applies the Krylov subspace method for efficient computation. Finally, Section 5.9 concludes with a proof of local superlinear/quadratic convergence of the prototype RIPM algorithm under applying the perturbed damped Riemannian Newton method for analysis.

Chapter 6

Global Convergent Algorithm of RIPM

Part	Section
Part 1. Core proposal	6.1 Interpretation of Algorithm 10 6.2 Sufficient Decreasing Condition 6.3 Centrality Conditions
Part 2. Core analysis	6.4 Global Convergence Theorem 6.5 Auxiliary I: Continuity of Some Special Scalar Fields 6.6 Auxiliary II: Boundedness of Sequences 6.7 Proofs of Global Convergence Theorem 20
Part 3. Experiments and summary	6.8 Numerical Experiments 6.9 Summary

In previous chapter, we discussed the prototype algorithm of Riemannian Interior Point Method (RIPM) in Algorithm 6 and gave its local convergent result. In this chapter, however, we will propose a globally convergent version of RIPM, which uses the classical line search described in [73]. We first describe the globally convergent algorithm of RIPM in Algorithm 10 as follows, and we will explain it step by step and prove its global convergent theorem in the following sections.

Algorithm 10: Global Convergent Algorithm of RIPM for (CRO)

Input: A problem of (CRO), an initial point $w_0 = (x_0, y_0, z_0, s_0) \in \mathcal{N}$ with $(z_0, s_0) > 0$ and a retraction R on $\mathcal{M}$. $\theta \in (0,1), \beta \in (0, 0.5], \gamma_{-1} \in (0.5, 1)$.

Output: Sequence $\{w_k\} \subset \mathcal{N}$ such that $\{w_k\} \to w^*$ and w^* satisfies the KKT conditions (5.12).

Set $k \to 0$;

while *Stopping criterion not satisfied* **do**

1. Choose $\sigma_k \in (0,1)$ and

$$\rho_k \in \left[z_k^T s_k/m, \|F(w_k)\|/\sqrt{m}\right]; \tag{6.1}$$

2. Obtain $\Delta w_k = (\Delta x_k, \Delta y_k, \Delta z_k, \Delta s_k) \in T_{w_k}\mathcal{N}$ by solving the following linear equation:

$$\nabla F(w_k)\Delta w_k = -F(w_k) + \sigma_k \rho_k \hat{e}; \tag{6.2}$$

3. Step size selection:

(3a) Centrality conditions: Set $\gamma_k \in (0.5, \gamma_{k-1})$ and $\bar{\alpha}_k = \min\{\alpha_k^I, \alpha_k^{II}\}$ from (6.8) ;

(3b) Sufficient decrease condition: Let $\alpha_k := \theta^t \bar{\alpha}_k$, where t is the smallest nonnegative integer such that α_k satisfies

$$\varphi(\bar{\mathrm{R}}_{w_k}(\alpha_k \Delta w_k)) - \varphi(w_k) \le \alpha_k \beta \langle \operatorname{grad} \varphi(w_k), \Delta w_k \rangle; \tag{6.3}$$

4. Compute the next point as $w_{k+1} := \bar{\mathrm{R}}_{w_k}(\alpha_k \Delta w_k)$;

5. $k \to k+1$;

end

6.1 Interpretation of Algorithm 10

In this section, we will provide new symbols and merit function needed in order to explain Algorithm 10. In contrast to the prototype Algorithm 6, the main difference in the global Algorithm 10 is in the choice of the step size. There is no major difference in solving the Newton equation to obtain the Newton direction. Then, the efficient approach to solving the Newton equation discussed in Section 5.8 remains applicable. Consequently, we often assume that the current Newton direction Δw_k is already known.

6.1.1 New Symbols

For a starting point $w_0 = (x_0, y_0, z_0, s_0) \in \mathcal{N} = \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$ with $(z_0, s_0) > 0$, define two real numbers as

$$\tau_1 := \frac{\min(Z_0 S_0 \mathbf{1})}{z_0^T s_0/m}, \quad \tau_2 := \frac{z_0^T s_0}{\|F(w_0)\|_w}. \tag{6.4}$$

Here, m is the number of inequities constraints in (CRO). Then we have $0 < \tau_1 \le 1$ and $0 < \tau_2 \le \sqrt{m}$.

For simplicity, we often omit the subscript of iteration count k. Given the current point $w = (x, y, z, s) \in \mathcal{N}$ and current (Newton) direction $\Delta w = (\Delta x, \Delta y, \Delta z, \Delta s) \in T_w\mathcal{N} \cong T_x\mathcal{M} \times \mathbb{R}^l \times$

$\mathbb{R}^m \times \mathbb{R}^m$, the next iterate is calculated along a curve on $\mathcal{N}$, namely,

$$\alpha \mapsto w(\alpha) := \bar{\mathrm{R}}_w(\alpha \Delta w), \tag{6.5}$$

for some step size $\alpha > 0$. Recall that the retraction $\bar{\mathrm{R}}$ on $\mathcal{N}$ is defined as (2.43). By introducing the notations

$$w(\alpha) \equiv (x(\alpha), y(\alpha), z(\alpha), s(\alpha)),$$

we have

$$\begin{aligned} &x(\alpha) \equiv \mathrm{R}_x(\alpha \Delta x) \in \mathcal{M}, \\ &y(\alpha) \equiv y + \alpha \Delta y,\ z(\alpha) \equiv z + \alpha \Delta z,\ s(\alpha) \equiv s + \alpha \Delta s. \end{aligned}$$

Define two real to real functions

$$\alpha \mapsto f^I(\alpha) := \min(Z(\alpha)S(\alpha)\mathbf{1}) - \gamma \tau_1 z(\alpha)^T s(\alpha)/m, \tag{6.6}$$

$$\alpha \mapsto f^{II}(\alpha) := z(\alpha)^T s(\alpha) - \gamma \tau_2 \|F(w(\alpha))\|_w, \tag{6.7}$$

where $\gamma \in (0,1)$ is a constant. Once the initial point w_0 is selected, then τ_1, τ_2 above are also constants. Here, the big letters $Z(\alpha), S(\alpha)$ represent the diagonal matrices whose diagonals are given by the vectors $z(\alpha), s(\alpha)$, respectively. Now, given the functions $f^i(\alpha)$, for $i = I, II$, furthermore compute

$$\alpha^i := \max_{\alpha \in (0,1]} \left\{ \alpha \mid f^i(t) \geq 0, \text{ for all } t \in (0, \alpha] \right\}, \tag{6.8}$$

i.e., α^i are either 1 or the smallest positive root for the functions $f^i(\alpha)$ in $(0,1]$. These are called *centrality conditions* and will be discussed more in Section 6.3 later.

Remark 22. We note that the functions $f^i(\alpha)$ (thus numbers α^i) for $i = I, II$ depend on the iteration count k, because the definition of curve $w(\alpha)$ in (6.5) is determined by current point w_k and current tangent vector Δw_k. For simplicity we often choose not to write explicitly this dependency.

Remark 23. For a moment, we allow the existence of numbers α^i for $i = I, II$ in (6.8). A rigorous proof will be presented in Proposition 14 later.

6.1.2 Merit Function

In constrained optimization, a merit function plays a crucial role by providing a mechanism to unify the objective function and constraint violations, thereby guiding iterative algorithms towards feasible and optimal solutions. Refer to [147, Section 15.4] for more. Choosing the Karush-Kuhn-Tucker (KKT) conditions as a basis for constructing a merit function in constrained optimization can be an intuitive choice. In previous Section 5.2, we know that the (Riemannian) KKT conditions (5.3)-(5.7) give a set of necessary conditions for a solution to be feasible and optimal for (CRO). Thus, in Algorithm 10, we selected the simplest merit function, $\varphi(w) := \|F(w)\|_w^2$ (subscript w is often omitted), which quantifies the violation of the Riemannian KKT conditions. And, the ultimate goal of Algorithm 10 is to find w^* such that $\varphi(w^*) = 0$ but while keeping z^*, s^* nonnegative, see (5.14).

Proposition 12 (Gradient of Merit Function)**.** *Consider the KKT vector field F in (5.12) of (CRO). Define a merit function $\varphi\colon \mathcal{N} \to \mathbb{R}$ by*

$$\varphi(w) := \|F(w)\|^2, \tag{6.9}$$

then we have

$$\operatorname{grad} \varphi(w) = 2\nabla F(w)^*[F(w)]. \tag{6.10}$$

Here, $\nabla F(w)^*$ is the adjoint operator of $\nabla F(w)$, see (5.21).

Proof. Equation (6.10) comes from the following derivation. Let $\xi \in T_w\mathcal{N}$ and choose an arbitrary $V \in \mathfrak{X}(\mathcal{N})$ with $V(w) = \xi$, we have

$$\begin{aligned}
\mathrm{D}\varphi(w)[\xi] &= \mathrm{D}\varphi(w)[V(w)] \\
&= (V\varphi)(w) \text{ (by (2.16), } V \text{ can be seem as a map } V\colon \mathfrak{F}(\mathcal{N}) \to \mathfrak{F}(\mathcal{N})) \\
&= (V\langle F, F\rangle)(w) \text{ (by } \varphi = \|F\|^2 = \langle F, F\rangle, \text{ also see (2.26))} \\
&= 2\left\langle \nabla_\xi F, F\right\rangle(w) \text{ (by Theorem 3, compatibility with the metric for } \nabla) \\
&= 2\langle \nabla F(w)[\xi], F(w)\rangle \text{ (by definition (2.28))} \\
&= \langle \xi, 2\nabla F(w)^*[F(w)]\rangle\,.
\end{aligned}$$

The assertion (6.10) now follows from definition of Riemannian gradient. □

We conclude this section with some observations. Since the norm on the tangent space of product manifold is given by (2.41), it is easy to see (recall (5.12)):

$$\|F(w)\|_w^2 = \|\operatorname{grad}_x \mathcal{L}(w)\|_x^2 + \|h(x)\|_2^2 + \|g(x) + s\|_2^2 + \|ZS\mathbf{1}\|_2^2. \tag{6.11}$$

Let $\|v\|_1 = \sum_{i=1}^m |v_i|$ be the l_1 norm for $v \in \mathbb{R}^m$. It is well-known that the relation between l_1 and l_2 norm is $\|v\|_2 \le \|v\|_1 \le \sqrt{m}\|v\|_2$ for all $v \in \mathbb{R}^m$. Then for any positive $z, s \in \mathbb{R}^m$, one has

$$\|ZS\mathbf{1}\|_2 \le z^T s = \|ZS\mathbf{1}\|_1 \le \sqrt{m}\,\|ZS\mathbf{1}\|_2\,. \tag{6.12}$$

Hence, dividing both sides by $\sqrt{m}$, and (6.11) leads to

$$\|ZS\mathbf{1}\|_2 / \sqrt{m} \le z^T s/\sqrt{m} \le \|ZS\mathbf{1}\|_2 \le \|F(w)\|. \tag{6.13}$$

This also shows that the range of choices for ρ_k in (6.1) is well-defined.

6.2 Sufficient Decreasing Condition

In this section, let us focus on the Step (3b) sufficient decrease condition in Algorithm 10. Fixing the current point w and current direction $\Delta w \in T_w\mathcal{N}$, we denote a real to real function $\alpha \mapsto \phi(\alpha)$ by

$$\phi(\alpha) := \varphi(\bar{\mathrm{R}}_w(\alpha\Delta w)), \tag{6.14}$$

where the right-hand φ is the merit function defined in (6.9). Note that $\phi(0) = \varphi(w)$. To emphasize the k-th iteration, we often rewrite $\operatorname{grad}\varphi(w_k)$ by $\operatorname{grad}\varphi_k$, and $\varphi(\bar{\mathrm{R}}_{w_k}(\alpha\Delta w_k))$ by $\phi_k(\alpha)$. We also use notations

$$\phi_k \equiv \phi_k(0) \equiv \varphi(w_k) \equiv \varphi_k. \tag{6.15}$$

It follows from the Definition 16 of retraction on manifold that the derivative of $\phi(\alpha)$ at $\alpha = 0$ is

$$\phi'(0) = \mathrm{D}\varphi(\bar{\mathrm{R}}_w(0))\left[\mathrm{D}\bar{\mathrm{R}}_w(0)[\Delta w]\right] = \mathrm{D}\varphi(w)[\Delta w] = \langle \operatorname{grad}\varphi(w), \Delta w\rangle.$$

Hence at k-th w_k and Δw_k, we have

$$\phi_k'(0) = \langle \operatorname{grad}\varphi_k, \Delta w_k\rangle,$$

and condition (6.3) is to say

$$\phi_k(\alpha_k) - \phi_k(0) \leq \alpha_k\beta\phi_k'(0), \tag{6.16}$$

which is exactly the same as the sufficient decreasing condition (i.e., Armijo condition) in Euclidean optimization. Finally, let us consider $\phi_k'(0)$ in the right-hand side of the above inequality; by [147, Lemma 3.1], if $\phi_k'(0) < 0$, then the backtracking loop of Step (3b) in Algorithm 10 is well-defined, i.e., it will terminate at finite loops. This leads to the following concept.

Definition 48 (Descent Direction)**.** We say that Δw_k is a descent direction for $\varphi(w)$ in (6.9) at w_k if $\langle \operatorname{grad}\varphi_k, \Delta w_k\rangle < 0$.

The next lemma shows the condition under which the Newton direction Δw_k generated by (6.2) ensures the descent of the merit function.

Lemma 21 (Condition of Descent Direction — Setting of Parameter ρ_k)**.** *Consider the Algorithm 10. If the direction Δw_k is the solution of equation (6.2), then*

$$\phi_k'(0) = \langle \operatorname{grad}\varphi(w_k), \Delta w_k\rangle = 2\left(-\|F(w_k)\|^2 + \sigma_k\rho_k z_k^T s_k\right). \tag{6.17}$$

In this case, Δw_k is a descent direction for $\varphi(w)$ at w_k if and only if $\rho_k < \|F(w_k)\|^2/\sigma_k z_k^T s_k$.

Proof. The iteration count k is omitted. Let direction Δw be given as the solution of system (6.2), then

$$\begin{aligned}
\langle \operatorname{grad}\varphi(w), \Delta w\rangle &= \langle 2\nabla F(w)^* F(w), \Delta w\rangle \text{ (by (6.10))} \\
&= 2\langle F(w), \nabla F(w)\Delta w\rangle \\
&= 2\langle F(w), -F(w) + \sigma\rho\hat{e}\rangle \text{ (by (6.2))} \\
&= 2(-\langle F(w), F(w)\rangle + \sigma\rho\langle F(w), \hat{e}\rangle) \\
&= 2(-\|F(w)\|^2 + \sigma\rho z^T s).
\end{aligned}$$

For last equality above, note that by definitions of F in (5.12) and $\hat{e}$ in (5.26), one has $\langle F(w), \hat{e}\rangle = \langle ZS\mathbf{1}, \mathbf{1}\rangle = z^T s$. Hence, $\phi'(0) = \langle \operatorname{grad}\varphi(w), \Delta w\rangle < 0$ if and only if $\rho < \|F(w)\|^2/\sigma z^T s$. □

In particular, our choice for ρ_k in (6.1) of Algorithm 10 satisfies the condition stated in Lemma 21, thus our Algorithm 10 always generates the descent directions Δw_k. We formalize this result in the next

proposition and also show that the Algorithm 10 can generate monotone nonincreasing sequences $\{\varphi_k\}$. Note that $\phi_{k+1} = \varphi(w_{k+1}) = \phi_k(\alpha_k)$ under the notations (6.15).

Proposition 13 (Monotonically Nonincreasing Sequence $\{\varphi_k\}$)**.** *If $\|F(w_k)\| \neq 0$, then the direction Δw_k generated by Algorithm 10 is a descent direction for the merit function $\varphi(w)$ at w_k. Moreover, if Armijo condition (6.3) is satisfied, then we have*

$$\phi_k(\alpha_k) \leq [1 - 2\alpha_k\beta(1-\sigma_k)]\,\phi_k(0).$$

And the sequence $\{\varphi_k\}$ is monotonically nonincreasing.

Proof. The iteration count k is omitted. Suppose that we choose $\rho \leq \|F(w)\|/\sqrt{m}$ as in (6.1) and Δw is given by (6.2), we have

$$\begin{aligned}
\phi'(0) =& \langle \operatorname{grad}\varphi(w), \Delta w\rangle \\
=& 2(-\varphi(w) + \sigma\rho z^T s) \text{ (by (6.17) and (6.9))} \\
\leq& 2(-\varphi(w) + \sigma\|F(w)\| z^T s/\sqrt{m}) \\
\leq& 2(-\varphi(w) + \sigma\|F(w)\|^2) \text{ (by (6.13))} \\
=& -2(1-\sigma)\varphi(w) \\
=& -2(1-\sigma)\phi(0) < 0. \text{ (since } \sigma \in (0,1) \text{ in Algorithm 10)}
\end{aligned} \tag{6.18}$$

This directly gives the result that Δw is a descending direction. Alternatively, if we use the condition from Lemma 21, it is sufficient to show that

$$\|F(w)\|/\sqrt{m} < \|F(w)\|^2/\sigma z^T s. \tag{6.19}$$

Note that by (6.13), $\sigma z^T s < z^T s \leq \sqrt{m}\|F(w)\|$; then, $\frac{1}{\sqrt{m}} < \frac{\|F(w)\|}{\sigma z^T s}$. Multiplying both sides by $\|F(w)\|$ gives (6.19).

Moreover, if condition (6.3), i.e., (6.16) is satisfied, then under the notations (6.15) we have

$$\begin{aligned}
\phi_{k+1} = \phi_k(\alpha_k) &\leq \phi_k(0) + \alpha_k\beta\phi_k'(0) \\
&\leq \phi_k(0) + \alpha_k\beta[-2(1-\sigma_k)\phi_k(0)] \text{ (by (6.18))} \\
&= [1 - 2\alpha_k\beta(1-\sigma_k)]\phi_k.
\end{aligned}$$

Finally, in Algorithm 10, note that we set $\beta \in (0, 1/2]$, $\sigma_k \in (0,1)$, and $\alpha_k \in (0,1]$, hence $0 < 1 - 2\alpha_k\beta(1-\sigma_k) < 1$, which implies that the sequence $\{\phi_k\}$ ($\{\varphi_k\}$) is monotonically nonincreasing. □

Remark 24. We can obtain the global Q-linear convergence of the values of the merit function φ_k to zero, if moreover, we request that $\{\alpha_k\}$ is bounded away from zero. Note that

$$\lim_{k\to\infty}\frac{\varphi_{k+1}}{\varphi_k} = \lim_{k\to\infty}\frac{\phi_{k+1}}{\phi_k} \leq \lim_{k\to\infty}[1 - 2\alpha_k\beta(1-\sigma_k)] < 1.$$

6.3 Centrality Conditions

In Remark 23, we assumed the existence of numbers α^i for $i = I, II$ in (6.8) for a moment. The next proposition shows that the strict feasibility of the initial vectors $z_0, s_0 > 0$ is sufficient to guarantee the existence.

Proposition 14 (Well-definedness of Centrality Conditions)**.** *Let $f^I(\alpha)$ and $f^{II}(\alpha)$ be the centrality functions defined in (6.6) and (6.7) where τ_1, τ_2 are given in (6.4). Set a sequence of parameters $\{\gamma_k\}$ with $0.5 < \cdots < \gamma_2 < \gamma_1 < \gamma_0 < 1$. If $z_0 > 0, s_0 > 0$, then for each $k = 0, 1, 2, \ldots$,*

(i) there exists $\alpha_k^I > 0$ such that $f_k^I(\alpha) \geq 0$, for all $\alpha \in (0, \alpha_k^I]$,

(ii) there exists $\alpha_k^{II} > 0$ such that $f_k^{II}(\alpha) \geq 0$, for all $\alpha \in (0, \alpha_k^{II}]$.

Hence, the Step (3a) centrality conditions in Algorithm 10 is well-defined.

Proof. We only prove (i) by mathematical induction. The same idea can be applied to (ii). Consider $k = 0$. By construct of τ_1 in (6.4), we have

$$\begin{aligned} f_0^I(0) &= \min(Z_0 S_0 \mathbf{1}) - \gamma_0 \tau_1 (z_0^T s_0/m) \\ &= \min(Z_0 S_0 \mathbf{1}) - \gamma_0 \min(Z_0 S_0 \mathbf{1}) \\ &= (1 - \gamma_0) \min(Z_0 S_0 \mathbf{1}) > 0. \text{ (since } z_0 > 0, s_0 > 0) \end{aligned}$$

Thus, $f_0^I(0) > 0$. It follows the continuity of f_0^I at $\alpha = 0$, there exists $\alpha_0^I > 0$ such that $f_0^I(\alpha) \geq 0, \forall \alpha \in (0, \alpha_0^I]$. Hence, (i) holds for $k = 0$.

Suppose that for $k = 0, 1, 2, \ldots$, (i) holds. Note that at k-th iteration, the quantities $w_k, \Delta w_k$ and γ_k together determine the function:

$$f_k^I(\alpha) = \min(Z_k(\alpha) S_k(\alpha) \mathbf{1}) - \gamma_k \tau_1 (z_k(\alpha)^T s_k(\alpha)/m).$$

A final step α_k is chosen such that $f_k^I(\alpha_k) \geq 0$. For $k + 1$, we have

$$\begin{aligned} f_{k+1}^I(0) &= \min(Z_{k+1}(0) S_{k+1}(0) \mathbf{1}) - \gamma_{k+1} \tau_1 (z_{k+1}(0)^T s_{k+1}(0)/m) \\ &= \min(Z_{k+1} S_{k+1} \mathbf{1}) - \gamma_{k+1} \tau_1 (z_{k+1}^T s_{k+1}/m) \\ &> \min(Z_{k+1} S_{k+1} \mathbf{1}) - \gamma_k \tau_1 (z_{k+1}^T s_{k+1}/m) \text{ (since } 0 < \gamma_{k+1} < \gamma_k) \\ &= \min(Z_k(\alpha_k) S_k(\alpha_k) \mathbf{1}) - \gamma_k \tau_1 (z_k(\alpha_k)^T s_k(\alpha_k)/m) \\ &= f_k^I(\alpha_k) \geq 0. \end{aligned}$$

Thus, $f_{k+1}^I(0) > 0$. By continuity of f_{k+1}^I at $\alpha = 0$, there exists $\alpha_{k+1}^I > 0$ such that $f_{k+1}^I(\alpha) \geq 0, \forall \alpha \in (0, \alpha_{k+1}^I]$. This completes the proof. □

The role of the centrality conditions is to ensure the nonnegativity of z_k and s_k during the iteration process (through f_k^I), as well as to prevent $z_k^T s_k$ from becoming too small (through f_k^{II}). Too small $z_k^T s_k$ may lead to too small selection of ρ_k, which may affect the efficiency of the iteration of interior point method. The above considerations are not any different from the interior point method in Euclidean optimization. See [73, 28, 70] for more.

At this point, all ingredients in Algorithm 10 have been explained and are well-defined. In the next section, we directly state the convergence theorem of Algorithm 10. We end this section with following minor result.

Lemma 22 (Bound of the Sequence $\{z_k^T s_k\}$)**.** *Let $\{w_k\}$ be generated by Algorithm 10 and $\chi := \tau_2^2/4$. Let m be the number of inequities constraints in (CRO). Then,*

$$\chi\varphi(w_k) \leq (z_k^T s_k)^2 \leq m\varphi(w_k).$$

Proof. By (6.7), we have $z_k^T s_k \geq \gamma_k \tau_2 \|F(w_k)\| \geq \tau_2/2\|F(w_k)\|$, then $(z_k^T s_k)^2 \geq (\tau_2/2)^2\varphi(w_k)$, which leads to the first inequality. The second inequality follows (6.12). □

6.4 Global Convergence Theorem

Given an initial point w_0 and $\varepsilon \geq 0$, let us define the set

$$\Omega(\varepsilon) := \{w \in \mathcal{N} : \varepsilon \leq \varphi(w) \leq \varphi_0, \min(ZS\mathbf{1})/(z^T s/m) \geq \tau_1/2,\ z^T s/\|F(w)\| \geq \tau_2/2\}.$$

Note that $\Omega(\varepsilon)$ is a closed subset of $\mathcal{N}$; $\Omega(\varepsilon) \subset \Omega(0)$ for any $\varepsilon > 0$. We now establish global convergence of the Algorithm 10 for (CRO) under the following assumptions.

Assumption 3 (Assumptions of Global Convergence of (CRO))**.**

(C1) in the set $\Omega(0)$, the functions f, h, g are smooth; the gradients of the equality constraints are linearly independent vector fields, i.e., the set $\{\operatorname{grad} h_j(x)\}_{j=1}^l$ is linearly independent in $T_x\mathcal{M}$ for all x; the map $w \mapsto \nabla F(w)$ is Lipschitz continuous (with respect to parallel transport);

(C2) the sequences $\{x_k\}$ and $\{z_k\}$ are bounded [71, 28];

(C3) in any compact subset of $\Omega(0)$, the operator $\nabla F(w)$ is nonsingular.

Given the above assumptions, we can now prove the following statement.

Theorem 20 (Global Convergence of Algorithm 10)**.** *Let $\{w_k\}$ be generated by Algorithm 10 with* $\mathrm{R} = \mathrm{Exp}$ *and $\{\sigma_k\} \subset (0, 1)$ be bounded away from zero and one. Let φ be Lipschitz continuous on $\Omega(0)$. If assumptions (C1)-(C3) hold, then $\{\|F(w_k)\|\}$ converges to zero; and for any limit point $w^* = (x^*, y^*, z^*, s^*)$ of $\{w_k\}$, x^* is a Riemannian KKT point of problem (CRO).*

Note that although the exponential map Exp is used in the theorem, the numerical experiments indicate that global convergence may hold for a general retraction R. The proof of above theorem will be given in the following three sections. We first discuss the two classes of auxiliary results in Section 6.5 and Section 6.5, and eventually gives the proofs in Section 6.7.

6.5 Auxiliary I: Continuity of Some Special Scalar Fields

Many functions in Euclidean setting have an obvious continuity that is important for convergence analysis. However, the counterparts of those functions in Riemannian setting lacks a direct result of their continuity. In this section we focus on this issue and record these results in Proposition 15 and Proposition 16.

Let $\mathcal{M}$ be a Riemannian manifold. If we assign a linear operator $\mathcal{A}_x \colon T_x\mathcal{M} \to T_x\mathcal{M}$ to each point $x \in \mathcal{M}$, then the map

$$x \mapsto \|\mathcal{A}_x\|$$

is a well-defined scalar field[1] on $\mathcal{M}$, and we should be aware that the operator norm $\|\cdot\|$ (see (2.1)) depends on x since the domain and codomain of $\mathcal{A}_x$ depend on x. Let $\|\cdot\|_2, \|\cdot\|_{\mathrm{F}}$ be the spectral norm, Frobenius norm of matrices, respectively. Notice that $\|\cdot\|_2$ is l_2 norm for vector argument and spectral norm for matrix argument. Next lemma shows that the maps

$$x \mapsto \|\hat{\mathcal{A}}_x\|_2 \ \text{ and } \ x \mapsto \|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$$

are also well-defined scalar fields, where $\hat{\mathcal{A}}_x$ denotes the matrix representation of $\mathcal{A}_x$ with respect to *arbitrary* orthonormal basis of $T_x\mathcal{M}$. The term "well-defined", here, means that the values $\|\hat{\mathcal{A}}_x\|_2$ and $\|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$ are invariant under a change of orthonormal basis.

Lemma 23 (Well-definedness of Two Special Scalar Fields)**.** *Let $\mathcal{M}$ be a d-dimensional Riemannian manifold endowed with a metric $\langle\cdot,\cdot\rangle$. Let $x \in \mathcal{M}$ and $\mathcal{A}_x$ be a linear operator from and to $T_x\mathcal{M}$. Choose a basis of $T_x\mathcal{M}$ that is orthonormal with respect to the inner product $\langle\cdot,\cdot\rangle_x$, and let $\hat{\mathcal{A}}_x \in \mathbb{R}^{d\times d}$ denote the matrix representation of $\mathcal{A}_x$ under the basis. Then, the values $\|\hat{\mathcal{A}}_x\|_2$ and $\|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$ are invariant under a change of orthonormal basis; moreover,*

$$\|\mathcal{A}_x\| = \|\hat{\mathcal{A}}_x\|_2 \le \|\hat{\mathcal{A}}_x\|_{\mathrm{F}}.$$

Proof. Suppose that there two orthonormal bases $\{E_i\}_{i=1}^d$, $\{E_i'\}_{i=1}^d$ on $T_x\mathcal{M}$. With respect to those bases, let $P \in \mathbb{R}^{d\times d}$ denote the change-of-basis matrix, i.e.,

$$[P]_{kj} := \langle E_j', E_k\rangle_x, \ \text{for } 1 \le k, j \le d.$$

It is known that P is an orthogonal matrix due to the orthonormal properties of $\{E_i\}_{i=1}^d$ and $\{E_i'\}_{i=1}^d$. Let $\hat{\mathcal{A}}_x, \hat{\mathcal{A}}_x' \in \mathbb{R}^{d\times d}$ denote the matrix representations of $\mathcal{A}_x$ with respect to the two bases, respectively. From [9, 10.7, P298], we have $\hat{\mathcal{A}}_x' = P^{-1}\hat{\mathcal{A}}_x P$. Since P is orthogonal,

$$\|\hat{\mathcal{A}}_x'\| = \|P^{-1}\hat{\mathcal{A}}_x P\| = \|\hat{\mathcal{A}}_x\|$$

hold for the Frobenius norm, or the spectral norm. Therefore, the values $\|\hat{\mathcal{A}}_x\|_2$ and $\|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$ are invariant under a change of orthonormal basis.

Now, choose an orthonormal basis $\{E_i\}_{i=1}^d$ on $T_x\mathcal{M}$. For any $y \in T_x\mathcal{M}$, its vector representation $\hat{y} \in \mathbb{R}^d$ is defined by $y = \sum_{i=1}^d \hat{y}_i E_i$. Thus, $y \mapsto \hat{y}$ establishes an isomorphism between $T_x\mathcal{M}$ and $\mathbb{R}^d$. Then we have $\widehat{\mathcal{A}_x y} = \hat{\mathcal{A}}_x\hat{y}$, i.e., $\mathcal{A}_x y = \sum_{i=1}^d (\hat{\mathcal{A}}_x\hat{y})_i E_i$, see [9, 3.65, P85]; and from the property of orthonormal basis (see [9, 6.25, P180]), we have

$$\|\mathcal{A}_x y\|_x^2 = \|\sum_{i=1}^d (\hat{\mathcal{A}}_x\hat{y})_i E_i\|_x^2 = \sum_{i=1}^d (\hat{\mathcal{A}}_x\hat{y})_i^2 = \|\hat{\mathcal{A}}_x\hat{y}\|_2^2. \tag{6.20}$$

[1] The term "scalar field" is equal to "real-valued functions".

Thus, $\|\mathcal{A}_x y\|_x = \|\hat{\mathcal{A}}_x \hat{y}\|_2$ for any $y \in T_x\mathcal{M}$. Finally, we obtain

$$\|\mathcal{A}_x\| = \sup_{y\in T_x\mathcal{M}, \|y\|_x=1} \{\|\mathcal{A}_x y\|_x\} = \sup_{\hat{y}\in\mathbb{R}^d, \|\hat{y}\|_2=1} \{\|\hat{\mathcal{A}}_x \hat{y}\|_2\} = \|\hat{\mathcal{A}}_x\|_2. \tag{6.21}$$

It is clear that $\|\hat{\mathcal{A}}_x\|_2 \leq \|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$ (see [59, Theorem 3.1.3]). □

From Lemma 23, we know that $\|\mathcal{A}_x\| = \|\hat{\mathcal{A}}_x\|_2$ holds for all $x \in \mathcal{M}$. Then $x \mapsto \|\hat{\mathcal{A}}_x\|_2$ and $x \mapsto \|\mathcal{A}_x\|$ are the same function. Thus, the continuity of $\|\hat{\mathcal{A}}_x\|_2$ implies continuity of $\|\mathcal{A}_x\|$. However, we do not have clarity on the continuity of scalar fields $x \mapsto \|\hat{\mathcal{A}}_x\|_2$ and/or $x \mapsto \|\hat{\mathcal{A}}_x\|_{\mathrm{F}}$ yet, since continuity depends on how $x \mapsto \mathcal{A}_x$. In our problem (CRO), we consider a special case of $x \mapsto \mathcal{A}_x$, that is, $x \mapsto \operatorname{Hess} f(x)$. The following proposition illustrates the continuity of such an important case.

Proposition 15 (Continuity of Some Special Scalar Fields - I)**.** *Consider smooth function f in (CRO) where $\mathcal{M}$ be a d-dimensional Riemannian manifold endowed with a metric $\langle\cdot,\cdot\rangle$. Let $\widehat{\operatorname{Hess} f(x)} \in \mathrm{S}(d)$ denote the matrix representation of $\operatorname{Hess} f(x)$ with respect to an arbitrary orthonormal basis of $T_x\mathcal{M}$. Then, the map*

$$x \mapsto \|\widehat{\operatorname{Hess} f(x)}\|$$

is a well-defined continuous scalar field on $\mathcal{M}$, for the Frobenius norm, or the spectral norm. Moreover,

$$x \mapsto \|\operatorname{Hess} f(x)\|$$

is a well-defined continuous scalar field on $\mathcal{M}$. The above results can be applied verbatim to the Hessian of constraint functions $\{h_j\}_{j=1}^{l}, \{g_i\}_{i=1}^{m}$ in (CRO).

Proof. Lemma 23 shows that scalar field $x \mapsto \|\widehat{\operatorname{Hess} f(x)}\|$ is well-defined, it suffices to prove its continuity. Recall Definition 18 of local frame in Section 2.8, for each $\bar{x} \in \mathcal{M}$ there is a smooth orthonormal local frame $\{E_i\}_{i=1}^{d}$ on a open neighborhood $\mathcal{U}$ of $\bar{x}$, namely, $\{E_1(x), \ldots, E_d(x)\}$ forms an orthonormal basis on $T_x\mathcal{M}$ for all $x \in \mathcal{U}$ (see [127, Corollary 13.8]). Choose such a local frame $\{E_i\}_{i=1}^{d}$ around $\bar{x}$, then the matrix representation of $\operatorname{Hess} f(x)$ with respect to $\{E_1(x), \ldots, E_d(x)\}$ is given by, for $1 \leq k, j \leq d$,

$$[\widehat{\operatorname{Hess} f(x)}]_{kj} := \langle \operatorname{Hess} f(x)[E_j(x)], E_k(x)\rangle_x = \langle (\nabla_{E_j} \operatorname{grad} f)(x), E_k(x)\rangle_x.$$

The last equality comes form Definition 23 of $\operatorname{Hess} f(x)$. Now, from the smoothness of Riemannian metric $\langle\cdot,\cdot\rangle$ (see (2.21)), it follows that $x \mapsto \widehat{\operatorname{Hess} f(x)}$ is a continuous function from $\mathcal{U} \subset \mathcal{M}$ to $\mathrm{S}(d)$. Since any matrix norm is continuous, $\|\widehat{\operatorname{Hess} f(x)}\|$ is continuous on $\mathcal{U}$ containing $\bar{x}$. Because the discussion above hold for any $\bar{x} \in \mathcal{M}$. We complete the proof. □

The proof of the next proposition is similar to that of Lemma 23 and Proposition 15. It gives the continuity of another special class of scalar fields.

Proposition 16 (Continuity of Some Special Scalar Fields - II)**.** *Consider smooth functions h, g in (CRO) and the linear operator $\mathcal{H}_x$ and $\mathcal{G}_x$ defined in (5.19). Then*

$$x \mapsto \|\mathcal{H}_x\| \text{ and } x \mapsto \|\mathcal{G}_x\|$$

are well-defined continuous scalar fields on $\mathcal{M}$.

Proof. We only prove the result for $\{h_j\}_{j=1}^l$, because it can be applied verbatim to $\{g_i\}_{i=1}^m$. Recall that $H_x\colon \mathbb{R}^l \to T_x\mathcal{M}$ is given by

$$H_x y := \sum_{j=1}^{l} y_j \operatorname{grad} h_j(x).$$

Since scalar field $x \mapsto \|\mathcal{H}_x\|$ is well-defined for each x, it suffices to prove the continuity. The main idea is the same as that of Lemma 23 and Proposition 15.

First we claim that $\|\mathcal{H}_x\| = \|\hat{\mathcal{H}}_x\|_2$ for each $x \in \mathcal{M}$, where $\hat{\mathcal{H}}_x$ denotes the matrix representation of $\mathcal{H}_x$ with respect to an arbitrary orthonormal basis of $T_x\mathcal{M}$ and standard basis $\{\mathbf{e}_j\}_{j=1}^l$ of $\mathbb{R}^l$. It is easy to see $\|\mathcal{H}_x y\|_x^2 = \|\hat{\mathcal{H}}_x y\|_2^2$ for all $y \in \mathbb{R}^l$. Then we have

$$\|\mathcal{H}_x\| = \sup_{y\in\mathbb{R}^l, \|y\|_2=1} \{\|\mathcal{H}_x y\|_x\} = \sup_{y\in\mathbb{R}^l, \|y\|_2=1} \{\|\hat{\mathcal{H}}_x y\|_2\} = \|\hat{\mathcal{H}}_x\|_2.$$

Next, since for each $\bar{x} \in \mathcal{M}$ there is a smooth, local orthonormal frame $\{E_i\}_{i=1}^d$ on a neighborhood $\mathcal{U}$ of $\bar{x}$, we choose such a local frame $\{E_i\}_{i=1}^d$ around $\bar{x}$. The matrix representation of $\mathcal{H}_x$ with respect to $\{E_i(x)\}_{i=1}^d$ and $\{\mathbf{e}_j\}_{j=1}^l$ is $\hat{\mathcal{H}}_x \in \mathbb{R}^{d\times l}$ given by, for $1 \le k \le d, 1 \le j \le l$,

$$[\hat{\mathcal{H}}_x]_{kj} = \langle \mathcal{H}_x \mathbf{e}_j, E_k(x)\rangle_x = \langle \operatorname{grad} h_j(x), E_k(x)\rangle_x . \tag{6.22}$$

Again, by the smoothness of Riemannian metric $\langle\cdot,\cdot\rangle$ (see (2.21)), it follows that $x \mapsto \hat{\mathcal{H}}_x$ is a continuous function from $\mathcal{U} \subset \mathcal{M}$ to $\mathbb{R}^{d\times l}$. Since any matrix norm is continuous, $\|\hat{\mathcal{H}}_x\|_2 = \|\mathcal{H}_x\|$ is continuous on $\mathcal{U}$ containing $\bar{x}$. Because the discussion above hold for any $\bar{x} \in \mathcal{M}$, we complete the proof. □

We end this section with next proposition that shows the matrix representation of covariant derivative of KKT Vector Field $\nabla F(w)$ given in (5.20). The result of Proposition 17 is rather trivial, but we give the proof for completeness.

Proposition 17 (Matrix Representation of Covariant Derivative of KKT Vector Field)**.** *Let $\mathcal{M}$ be a d-dimensional Riemannian manifold endowed with a metric $\langle\cdot,\cdot\rangle$. Given any $w = (x, y, z, s) \in \mathcal{N} = \mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$, consider the linear operator $\nabla F(w)\colon T_w\mathcal{N} \to T_w\mathcal{N}$ given in (5.20). Let $\{E_i\}_{i=1}^d$ be an orthonormal basis of $T_x\mathcal{M}$ and $\{\mathbf{e}_j\}_{j=1}^l$, $\{\mathbf{e}_i\}_{i=1}^m$ be the standard bases of $\mathbb{R}^l, \mathbb{R}^m$, respectively (should be clear in context). If we choose an orthonormal basis of $T_w\mathcal{N} = T_x\mathcal{M} \times \mathbb{R}^l \times \mathbb{R}^m \times \mathbb{R}^m$ (thus, $\dim T_w\mathcal{N} = d + l + 2m$) in the form of*

$$\{(E_i, 0, 0, 0)\}_{i=1}^d \cup \{(0_x, \mathbf{e}_j, 0, 0)\}_{j=1}^l \cup \{(0_x, 0, \mathbf{e}_i, 0)\}_{i=1}^m \cup \{(0_x, 0, 0, \mathbf{e}_i)\}_{i=1}^m . \tag{6.23}$$

Then the matrix representation of $\nabla F(w)$ is given by

$$\widehat{\nabla F(w)} = \begin{bmatrix} Q & B & C & 0 \\ B^T & 0 & 0 & 0 \\ C^T & 0 & 0 & I \\ 0 & 0 & S & Z \end{bmatrix},$$

i.e., a matrix of order $(d+l+2m)$ *and, where*

$Q := Q(w) := \widehat{\mathrm{Hess}_x \mathcal{L}(w)} \in \mathrm{S}(d)$ *is given by* $[Q]_{kj} := \langle \mathrm{Hess}_x \mathcal{L}(w)\,[E_j]\,, E_k\rangle_x$ *for* $1 \le k, j \le d$*;*

$B := B(x) = [\widehat{\mathrm{grad}\, h_1}(x), \cdots, \widehat{\mathrm{grad}\, h_l}(x)] \in \mathbb{R}^{d\times l}$;

$C := C(x) = [\widehat{\mathrm{grad}\, g_1}(x), \cdots, \widehat{\mathrm{grad}\, g_m}(x)] \in \mathbb{R}^{d\times m}$; *and the "hat" in* B, C *means corresponding vector representation under the basis* $\{E_i\}_{i=1}^d$ *of* $T_x\mathcal{M}$.

In this case, there is a continuous scalar field $T : \mathcal{N} \to \mathbb{R}$ *such that* $\|Q(w)\|_{\mathrm{F}} \le T(w)$ *for any* w*. Moreover,* $x \mapsto \|B(x)\|_{\mathrm{F}}$ *and* $x \mapsto \|C(x)\|_{\mathrm{F}}$ *are continuous scalar fields on* $\mathcal{M}$.

Proof. The matrix representation $\widehat{\nabla F(w)}$ under the basis (6.23) is obtained by a trivial process, so we will omit its description. From (5.9), we have the linear dependence between the Hessian operators: $\mathrm{Hess}_x \mathcal{L}(w) = \mathrm{Hess}\, f(x) + \sum_{j=1}^l y_j \,\mathrm{Hess}\, h_j(x) + \sum_{i=1}^m z_i \,\mathrm{Hess}\, g_i(x)$. By the linearity of matrix representation (see [9, 3.36 & 3.38, P73]), we have

$$Q(w) = \widehat{\mathrm{Hess}_x \mathcal{L}(w)} = \widehat{\mathrm{Hess}\, f}(x) + \sum_{j=1}^{l} y_j \widehat{\mathrm{Hess}\, h_j}(x) + \sum_{i=1}^{m} z_i \widehat{\mathrm{Hess}\, g_i}(x),$$

under the same basis $\{E_i\}_{i=1}^d$ of $T_x\mathcal{M}$. Thus,

$$\|Q(w)\|_{\mathrm{F}} \le \left(\|\widehat{\mathrm{Hess}\, f}(x)\|_{\mathrm{F}} + \sum_{j=1}^{l} |y_j| \|\widehat{\mathrm{Hess}\, h_j}(x)\|_{\mathrm{F}} + \sum_{i=1}^{m} |z_i| \|\widehat{\mathrm{Hess}\, g_i}(x)\|_{\mathrm{F}} \right) =: T(w).$$

Note that $\|Q(w)\|_{\mathrm{F}}$ is invariant to under a change of orthonormal basis of $T_x\mathcal{M}$, by Lemma 23. From Proposition 15, $\|\widehat{\mathrm{Hess}\, f}(x)\|_{\mathrm{F}}$, $\{\|\widehat{\mathrm{Hess}\, h_j}(x)\|_{\mathrm{F}}\}_{j=1}^l$, $\{\|\widehat{\mathrm{Hess}\, g_i}(x)\|_{\mathrm{F}}\}_{i=1}^m$ are all continuous function with respect to variable x. It follows from the construct of $T(w)$ that T is continuous with respect to $w = (x, y, z, s) \in \mathcal{N}$. As for $\|B(x)\|_{\mathrm{F}}$, since the basis $\{E_i\}_{i=1}^d$ is orthonormal ([9, 6.25, P180]), we have

$$\|B(x)\|_{\mathrm{F}}^2 = \sum_{j=1}^{l} \|\widehat{\mathrm{grad}\, h_j}(x)\|_2^2 = \sum_{j=1}^{l} \|\mathrm{grad}\, h_j(x)\|_x^2\,,$$

which implies the continuity of $\|B(x)\|_{\mathrm{F}}$ by discussion of (2.24). The same argument is valid for $\|C(x)\|_{\mathrm{F}}$, and we complete the proof. □

6.6 Auxiliary II: Boundedness of Sequences

In this section, we continue to show more auxiliary results that will eventually be used to prove the global convergence theorem. This section contains only Proposition 18 and Lemma 24, both of which assume that the sequence $\{\|F(w_k)\|\}$ does not converge to zero, i.e., there exists $\varepsilon > 0$ such that $w_k \in \Omega(\varepsilon)$ for all k. In the next section, when we prove the global convergence by contradiction, those auxiliary results will be very useful.

Proposition 18 (Boundedness of Some Sequences)**.** *Let assumptions (C1)-(C3) hold and* $\{w_k\}$ *be a sequence generated by Algorithm 10. If for some* $\varepsilon > 0$ *and* $w_k \in \Omega(\varepsilon)$ *for all* $k = 0, 1, 2, \ldots$*, then*

(a) the sequence $\{z_k^T s_k\}$ *and* $\{(z_k)_i (s_k)_i\}\,, i = 1, 2, \ldots, m$*, are all bounded above and below away from zero.*

(b) the sequence $\{z_k\}$ and $\{s_k\}$ are bounded above and component-wise bounded away from zero;
(c) the sequence $\{w_k\}$ is bounded;
(d) the sequence $\{\|\nabla F(w_k)^{-1}\|\}$ is bounded;
(e) the sequence $\{\Delta w_k\}$ is bounded.

Proof. **(a)** It follows from Lemma 22 that

$$0 < \chi\varepsilon \leq \chi\varphi(w_k) \leq (z_k^T s_k)^2 \leq m\varphi(w_k) \leq m\varphi_0.$$

Thus, the sequence $\{z_k^T s_k\}$ is bounded above and below away from zero. (6.12) implies that $\{(z_k)_i(s_k)_i\}$, for $i = 1, 2, \ldots, m$, is bounded above, since

$$(z_k)_i(s_k)_i \leq \|Z_k S_k \mathbf{1}\|_2 \leq z_k^T s_k.$$

The centrality function (6.6) and condition (6.8) give

$$(z_k)_i(s_k)_i \geq \min(Z_k S_k \mathbf{1}) \geq \frac{1}{2}\tau_1(z_k^T s_k/m).$$

Therefore, $\{(z_k)_i(s_k)_i\}$, for $i = 1, 2, \ldots, m$, are bounded below away from zero.

(b) The boundedness of $\{x_k\}$ implies that $\{\|g(x_k)\|\}$ is bounded above say, by M_2. Then we have

$$\|s_k\| \leq \|g(x_k) + s_k\| + \|-g(x_k)\| \leq \sqrt{\varphi_0} + M_2,$$

which shows that $\{s_k\}$ is bounded above.

Since $\{(z_k)_i(s_k)_i\}$ are bounded away from zero and $\{s_k\}$ is bounded above, it follows that $\{z_k\}$ is bounded away from zero. By contradiction, suppose that $\liminf z_k = 0$, i.e., there is a subsequence $z_{k_l} \to 0$. Since $\{s_k\}$ is bounded above, we have $(z_{k_l})_i(s_{k_l})_i \to 0$, for $i = 1, 2, \ldots, m$, which is a contradiction. Analogously, for the same argument, $\{s_k\}$ is bounded below away from zero, because $\{z_k\}$ is bounded above by assumption (C2).

(c) Based on the previous result (b), it suffices to prove that $\{y_k\}$ is bounded. For a moment, the iteration count k is omitted. Note that by (5.2),

$$\sum_{j=1}^{l} y_j \operatorname{grad} h_j(x) = \operatorname{grad}_x \mathcal{L}(w) - \operatorname{grad} f(x) - \sum_{i=1}^{m} z_i \operatorname{grad} g_i(x). \tag{6.24}$$

By using the notations $\mathcal{H}_x$ and $\mathcal{G}_x$ defined by (5.19), we rewrite (6.24) as

$$\mathcal{H}_x y = \operatorname{grad}_x \mathcal{L}(w) - \operatorname{grad} f(x) - \mathcal{G}_x z =: b. \tag{6.25}$$

By (C1), for all x, $\{\operatorname{grad} h_1(x), \cdots, \operatorname{grad} h_l(x)\}$ is linearly independent. Thus, $\operatorname{rank} \mathcal{H}_x = \dim \mathbb{R}^l$, i.e., $\mathcal{H}_x$ is injection. Then there exists the unique solution of $\mathcal{H}_x y = b$. From (6.25), we have

$$y = \left[(\mathcal{H}_x^* \mathcal{H}_x)^{-1} \mathcal{H}_x^*\right] (\operatorname{grad}_x \mathcal{L}(w) - \operatorname{grad} f(x) - \mathcal{G}_x z)\,. \tag{6.26}$$

Define $C_x\colon T_x\mathcal{M} \to \mathbb{R}^l$ by $C_x := \left(\mathcal{H}_x^*\mathcal{H}_x\right)^{-1}\mathcal{H}_x^*$. Under the orthonormal basis of $T_x\mathcal{M}$ and standard basis $\{\mathbf{e}_i\}_{i=1}^l$ of $\mathbb{R}^l$, if $\hat{\mathcal{H}}_x$ is the matrix corresponding to $\mathcal{H}_x$, then $\hat{C}_x = (\hat{\mathcal{H}}_x^T\hat{\mathcal{H}}_x)^{-1}\hat{\mathcal{H}}_x^T$ is the matrix corresponding to C_x [9, 7.10, P208; 3.43, P75]. Using the same idea as the context of equation (6.20), (6.21), we have $\|C_x\| = \|\hat{C}_x\|_2$ for any x. Recall that in the proof of Proposition 16, (6.22) shows that for each $\bar{x} \in \mathcal{M}$ there is a neighborhood $\mathcal{U}$ of $\bar{x}$ such that $x \mapsto \hat{\mathcal{H}}_x$ is continuous over $\mathcal{U}$. Then by function composition, $x \mapsto \hat{C}_x = (\hat{\mathcal{H}}_x^T\hat{\mathcal{H}}_x)^{-1}\hat{\mathcal{H}}_x^T$ is also continuous over $\mathcal{U}$. It shows that $\|C_x\| = \|\hat{C}_x\|_2$ is continuous at each point $\bar{x}$, hence, on $\mathcal{M}$.

Finally, with Proposition 16 together, $\|C_x\|, \|\mathrm{grad}\, f(x)\|, \|\mathcal{G}_x\|$ are all continuous on $\mathcal{M}$. Because $\{x_k\}$ is bounded, by (6.26) we have

$$\|y_k\| \le \|C_{x_k}\| \left(\|\mathrm{grad}_x\, \mathcal{L}(w_k)\| + \|\mathrm{grad}\, f(x_k)\| + \|G_{x_k}\|\, \|z_k\|\right) \le c_1\left(\sqrt{\varphi_0} + c_2 + c_3\, \|z_k\|\right),$$

for some positive constants c_1, c_2, c_3. Then $\{y_k\}$ is bounded because $\{z_k\}$ is bounded.

(d) For each w_k, choose an arbitrary orthonormal basis of $T_{w_k}\mathcal{N}$. If matrix $\widehat{\nabla F(w_k)}$ corresponds to $\nabla F(w_k)$, then the inverse matrix $\widehat{\nabla F(w_k)}^{-1}$ corresponds to $\nabla F(w_k)^{-1}$. By Lemma 23, we have $\left\|\nabla F(w_k)^{-1}\right\| \le \|\widehat{\nabla F(w_k)}^{-1}\|_{\mathrm{F}}$, where $\|\widehat{\nabla F(w_k)}^{-1}\|_{\mathrm{F}}$ is independent to the orthonormal basis we choose. Since sequence $\{\|\widehat{\nabla F(w_k)}^{-1}\|_{\mathrm{F}}\}$ is well-defined, it is sufficient to show that $\{\|\widehat{\nabla F(w_k)}^{-1}\|_{\mathrm{F}}\}$ is bounded.

For convenience, we choose the orthonormal basis of $T_{w_k}\mathcal{N}$ given in (6.23). Then, we have

$$\widehat{\nabla F(w_k)} = \begin{bmatrix} Q_k & B_k & C_k & 0 \\ B_k^T & 0 & 0 & 0 \\ C_k^T & 0 & 0 & I \\ 0 & 0 & S_k & Z_k \end{bmatrix}. \tag{6.27}$$

By Proposition 17, there is a continuous scalar field $T(w)$ on $\mathcal{N}$ such that $\|Q(w)\|_{\mathrm{F}} \le T(w)$; and $\|B(x)\|_{\mathrm{F}}$, $\|C(x)\|_{\mathrm{F}}$ are continuous on $\mathcal{M}$. It follows from the boundedness of $\{x_k\}$, $\{w_k\}$ that for all k,

$$\begin{aligned} \|Q_k\|_{\mathrm{F}} &= \|Q(w_k)\|_{\mathrm{F}} \le T(w_k) \le c_4, \\ \|B_k\|_{\mathrm{F}} &= \|B(x_k)\|_{\mathrm{F}} \le c_5, \\ \|C_k\|_{\mathrm{F}} &= \|C(x_k)\|_{\mathrm{F}} \le c_6, \end{aligned}$$

for some positive constants c_4, c_5, c_6.

Note that whichever basis $\{E_i\}_{i=1}^d$ is used in the form of (6.23), the structure of matrix $\widehat{\nabla F(w_k)}$ and the properties of its submatrix blocks in (6.27) remain unchanged, e.g., symmetry of Q_k; full-rank of B_k; identity matrix I in third row; all zero matrices; diagonal matrices S_k, Z_k; etc. This ensures that we can obtain the result by performing the appropriate decomposition of the matrix $\widehat{\nabla F(w_k)}$ as in the proof of Euclidean version. Up to this point, we have created all the conditions needed in the proof of Euclidean version. Applying [28, Theorem 2, (c)] directly, we claim that $\{\|\widehat{\nabla F(w_k)}^{-1}\|_{\mathrm{F}}\}$ is bounded.

(e) By (6.2), we have $\|\Delta w_k\| \le \|\nabla F(w_k)^{-1}\| \left(\|F(w_k)\| + \|\sigma_k\rho_k\hat{e}\|\right)$. We complete the proof. □

Lemma 24 (Boundedness of the Sequences $\{\bar{\alpha}_k\}$)**.** *Let assumptions (C1)-(C3) hold and $\{w_k\}$ be a sequence generated by Algorithm 10. If for some $\varepsilon > 0$ and $w_k \in \Omega(\varepsilon)$ for all $k = 0, 1, 2, \ldots$, $\{\sigma_k\}$ is bounded away from zero, and ρ_k is as in (6.1). Then, the sequence $\{\bar{\alpha}_k\}$ is bounded away from zero.*

Proof. Since $\bar{\alpha}_k = \min\{\alpha_k^I, \alpha_k^{II}\}$, it is sufficient to show that $\{\alpha_k^I\}$ and $\{\alpha_k^{II}\}$ are bounded away from zero. Let us suppress the iteration subscript k. First we have the following observations. In the perturbed Newton equation (6.2), we have $Z\Delta s + S\Delta z = \sigma\rho\mathbf{1} - ZS\mathbf{1}$, that is, for $i = 1, 2, \cdots, m$,

$$z_i\Delta s_i + s_i\Delta z_i = \sigma\rho - z_i s_i. \tag{6.28}$$

Summing up the items from $i = 1$ to m gives

$$\sum_{i=1}^{m}(z_i\Delta s_i + s_i\Delta z_i) = z^T\Delta s + s^T\Delta z = \sigma\rho m - z^T s. \tag{6.29}$$

Recall that $z(\alpha) = z + \alpha\Delta z$ and $s(\alpha) = s + \alpha\Delta s$. Let $z_i(\alpha)$ and $s_i(\alpha)$ denote their components, namely, $z(\alpha) = z_i + \alpha\Delta z_i$, $s_i(\alpha) = s_i + \alpha\Delta s_i$.

(Part 1) For result about α^I, see [73, Lemma 6.3], [67, Theorem 3.1]. The proofs in those references apply verbatim to the Riemannian case. However, their proofs are not complete, because of the lack of a strict interpretation for $\alpha^I \geq \alpha'$, that will be shown later.

It follows from the definition of f^I in (6.6) that

$$f^I(\alpha) = \min_i \left\{ z_i(\alpha)s_i(\alpha) - \frac{\gamma\tau_1}{m} z(\alpha)^T s(\alpha) \right\}.$$

Note that for $i = 1, 2, \cdots, m$,

$$\begin{aligned}
&z_i(\alpha)s_i(\alpha) - \frac{\gamma\tau_1}{m} z(\alpha)^T s(\alpha) \\
=&z_i s_i + \alpha\left(s_i\Delta z_i + z_i\Delta s_i\right) + \alpha^2\Delta z_i\Delta s_i - \frac{\gamma\tau_1}{m}\left[z^T s + \alpha\left(s^T\Delta z + z^T\Delta s\right) + \alpha^2\Delta z^T\Delta s\right] \\
=&\alpha^2\left(\Delta z_i\Delta s_i - \frac{\gamma\tau_1}{m}\Delta z^T\Delta s\right) + z_i s_i - \frac{\gamma\tau_1}{m} z^T s + \alpha\left[s_i\Delta z_i + z_i\Delta s_i - \frac{\gamma\tau_1}{m}\left(s^T\Delta z + z^T\Delta s\right)\right] \\
=&\alpha^2\left(\Delta z_i\Delta s_i - \frac{\gamma\tau_1}{m}\Delta z^T\Delta s\right) + z_i s_i - \frac{\gamma\tau_1}{m} z^T s + \alpha\left[\sigma\rho - z_i s_i - \gamma\tau_1\sigma\rho + \frac{\gamma\tau_1}{m} z^T s\right] \\
&\qquad \text{(by (6.28), (6.29))} \\
=&(1-\alpha)\left(z_i s_i - \frac{\gamma\tau_1}{m} z^T s\right) + \alpha(1-\gamma\tau_1)\sigma\rho + \alpha^2\left(\Delta z_i\Delta s_i - \frac{\gamma\tau_1}{m}\Delta z^T\Delta s\right) \\
\geq&\alpha(1-\gamma\tau_1)\sigma\rho - \alpha^2\left|\Delta z_i\Delta s_i - \frac{\gamma\tau_1}{m}\Delta z^T\Delta s\right|.
\end{aligned}$$

The last inequality above comes from the following result. For $\alpha \in (0, 1]$ and $x_1 \geq 0, x_2, x_2 \in \mathbb{R}$, one has

$$(1-\alpha)x_1 + \alpha x_2 + \alpha^2 x_3 \geq \alpha x_2 - \alpha^2|x_3|, \tag{6.30}$$

where in our cases, $x_1 := z_i s_i - \frac{\gamma\tau_1}{m} z^T s \geq \min_i\left\{z_i s_i - \frac{\gamma\tau_1}{m} z^T s\right\} = f^I(0) \geq 0$. Hence, from the boundedness of Δw, we have for $i = 1, 2, \cdots, m$,

$$z_i(\alpha)s_i(\alpha) - \frac{\gamma\tau_1}{m} z(\alpha)^T s(\alpha) \geq \alpha(1-\gamma\tau_1)\sigma\rho - \alpha^2 M_1 =: \psi_1(\alpha),$$

where ψ_1 is a quadratic function and M_1 is a constant such that

$$\left|\Delta z_i \Delta s_i - \frac{\gamma\tau_1}{m}\Delta z^T \Delta s\right| \leq M_1, \quad \forall i = 1, 2, \ldots, m.$$

It shows that the function ψ_1 is a lower bound of f^I on $\alpha \in (0, 1]$. Let $\alpha' := \frac{1-\gamma\tau_1}{M_1}\sigma\rho$ denote the nonzero root of $\psi_1(\alpha)$. Because $\gamma \in (1/2, 1)$, $\sigma \in (0, 1)$, and $\tau_1 \in (0, 1]$ is a constant that does not depend on k, and (6.1), we have

$$\alpha' < \frac{1-\tau_1/2}{M_1}\rho \leq \frac{(1-\tau_1/2)\left\|F(w)\right\|/\sqrt{m}}{M_1} \leq \frac{[(1-\tau_1/2)\varphi_0/\sqrt{m}]}{M_1},$$

which imply that we can take a sufficiently large M_1 so that the nonzero root α' is less than one, if necessary. Thus, by definition of α^I in (6.8), it follows that $\alpha^I \geq \alpha'$.

On the other hand, we know that

$$\alpha' = \frac{1-\gamma\tau_1}{M_1}\sigma\rho \geq \frac{1-\tau_1}{M_1}\sigma\rho \geq \left(\frac{1-\tau_1}{M_1 m}\right)\sigma z^T s.$$

Since $w \in \Omega(\varepsilon), \varepsilon > 0$, then $z^T s$ is bounded away from zero. Moreover, $\{\sigma_k\}$ is bounded away from zero; then, α^I is bounded away from zero.

(Part 2) Now, we show that α^{II} is bounded away from zero. Recall that $w(\alpha) = \bar{\mathrm{Exp}}_w(\alpha\Delta w)$. Fix α and let P_γ be the parallel transport along the geodesic $c(t) = \bar{\mathrm{Exp}}_w(t\alpha\Delta w)$. By the fundamental theorem of calculus in the Riemannian case, we obtain

$$\begin{aligned}
\mathrm{P}_c^{0\to 1}F(w(\alpha)) =& F(w) + \int_0^1 \mathrm{P}_c^{0\to t}\nabla F(c(t))\mathrm{P}_c^{t\to 0}\alpha\Delta w \mathrm{d}t \\
=& F(w) + \alpha\nabla F(w)\Delta w - \alpha\nabla F(w)\Delta w + \int_0^1 \mathrm{P}_c^{0\to t}\nabla F(c(t))\mathrm{P}_c^{t\to 0}\alpha\Delta w \mathrm{d}t \\
=& F(w) + \alpha\left(\sigma\rho\hat{e} - F(w)\right) + \alpha\int_0^1 \left[\mathrm{P}_c^{0\to t}\nabla F(c(t))\mathrm{P}_c^{t\to 0} - \nabla F(w)\right]\Delta w \mathrm{d}t \text{ (by (6.2))} \\
=& (1-\alpha)F(w) + \alpha\sigma\rho\hat{e} + \alpha\int_0^1 \left[\mathrm{P}_c^{0\to t}\nabla F(c(t))\mathrm{P}_c^{t\to 0} - \nabla F(w)\right]\Delta w \mathrm{d}t.
\end{aligned}$$

Taking the norm on both sides above gives

$$\begin{aligned}
&\left\|\mathrm{P}_c^{0\to 1}F(w(\alpha))\right\| \\
=& \left\|F(w(\alpha))\right\| \text{ (since parallel transport is isometric)} \\
\leq& (1-\alpha)\left\|F(w)\right\| + \alpha\sigma\rho\left\|\hat{e}\right\| + \alpha\int_0^1 \left\|\mathrm{P}_c^{0\to t}\nabla F(c(t))\mathrm{P}_c^{t\to 0} - \nabla F(w)\right\|\left\|\Delta w\right\| \mathrm{d}t \\
\leq& (1-\alpha)\left\|F(w)\right\| + \alpha\sigma\rho\sqrt{m} + \alpha\int_0^1 L\|t\alpha\Delta w\|\left\|\Delta w\right\| \mathrm{d}t \text{ (by Lipschitz continuity of } \nabla F) \\
=& (1-\alpha)\left\|F(w)\right\| + \alpha\sigma\rho\sqrt{m} + \frac{L}{2}\alpha^2\|\Delta w\|^2. \qquad (6.31)
\end{aligned}$$

From the above inequalities and definition of f^{II} in (6.7), it follows that

$$\begin{aligned}
f^{II}(\alpha) =& z(\alpha)^T s(\alpha) - \gamma\tau_2 \|F(w(\alpha))\| \\
=& z^T s + \alpha\left(s^T \Delta z + z^T \Delta s\right) + \alpha^2 \Delta z^T \Delta s - \gamma\tau_2 \|F(w(\alpha))\| \\
=& z^T s + \alpha\left(\sigma\rho m - z^T s\right) + \alpha^2 \Delta z^T \Delta s - \gamma\tau_2 \|F(w(\alpha))\| \text{ (by (6.29))} \\
=& (1-\alpha) z^T s + \alpha\sigma\rho m + \alpha^2 \Delta z^T \Delta s - \gamma\tau_2 \|F(w(\alpha))\| \\
\geq& (1-\alpha) z^T s + \alpha\sigma\rho m + \alpha^2 \Delta z^T \Delta s - \gamma\tau_2 \left[(1-\alpha)\|F(w)\| + \alpha\sigma\rho\sqrt{m} + \frac{L}{2}\alpha^2 \|\Delta w\|^2\right] \\
=& (1-\alpha)\left(z^T s - \gamma\tau_2 \|F(w)\|\right) + \alpha\sigma\rho\left(m - \gamma\tau_2\sqrt{m}\right) + \alpha^2\left(\Delta z^T \Delta s - \gamma\tau_2 \frac{L}{2}\|\Delta w\|^2\right) \\
\geq& \alpha\sigma\rho\sqrt{m}\left(\sqrt{m} - \gamma\tau_2\right) - \alpha^2 \left|\Delta z^T \Delta s - \gamma\tau_2 \frac{L}{2}\|\Delta w\|^2\right|.
\end{aligned}$$

The last inequality also comes from (6.30), where $x_1 := z^T s - \gamma\tau_2 \|F(w)\| = f^{II}(0) \geq 0$. Again, from the boundedness of Δw, we have

$$f^{II}(\alpha) \geq \alpha\sigma\rho\sqrt{m}\left(\sqrt{m} - \gamma\tau_2\right) - \alpha^2 M_2 =: \psi_2(\alpha),$$

where ψ_2 is a quadratic function and M_2 is a constant such that

$$\left|\Delta z^T \Delta s - \gamma\tau_2 \frac{L}{2}\|\Delta w\|^2\right| \leq M_2.$$

It shows that the function ψ_2 is a lower bound of f^{II} on $\alpha \in (0, 1]$. Let $\alpha'' := \frac{\sqrt{m}(\sqrt{m}-\gamma\tau_2)}{M_2}\sigma\rho$ denote the nonzero root of $\psi_2(\alpha)$. Similar to that of part (1), but note that constant $0 < \tau_2 \leq \sqrt{m}$, we have

$$\alpha'' < \frac{\sqrt{m}\left(\sqrt{m} - \tau_2/2\right)}{M_2}\rho \leq \frac{\|F(w)\|\left(\sqrt{m} - \tau_2/2\right)}{M_2} \leq \frac{\varphi_0\left(\sqrt{m} - \tau_2/2\right)}{M_2}.$$

which imply that we can take a sufficiently large M_2 so that $\alpha'' < 1$. Thus, by definition of α^{II} in (6.8), it follows that $\alpha^{II} \geq \alpha''$. On the other hand,

$$\alpha'' = \frac{\sqrt{m}\left(\sqrt{m} - \gamma\tau_2\right)}{M_2}\sigma\rho \geq \frac{\sqrt{m}\left(\sqrt{m} - \tau_2\right)}{M_2}\sigma\rho \geq \left(\frac{\sqrt{m} - \tau_2}{\sqrt{m}M_2}\right)\sigma z^T s.$$

Under the same reason as part (1), α^{II} is bounded away from zero. We complete the proof. □

6.7 Proofs of Global Convergence Theorem 20

In the last two sections, we have completed the required auxiliary results. We are ready to give the proof of the global convergence theorem.

Proof of Theorem 20. By Proposition 13, we know that $\{\|F(w_k)\|\}$ is monotonically nonincreasing, hence convergent. Assume that $\{\|F(w_k)\|\}$ does not converge to zero. Then, there exists $\varepsilon > 0$ such that

$\{w_k\} \subset \Omega(\varepsilon)$ for infinitely many k. We will show that the following two cases both lead to contradictions and thus the hypothesis $\|F(w_k)\| \nrightarrow 0$ is not valid.

(Case 1) For infinitely many k, if Step (3b) in Algorithm 10 is executed with $\alpha_k \equiv \bar{\alpha}_k$, it follows from Proposition 13 that

$$\varphi(w_{k+1})/\varphi(w_k) \leq [1 - 2\bar{\alpha}_k\beta(1-\sigma_k)].$$

Since $\{\bar{\alpha}_k\}$ is bounded away from zero by Lemma 24 and $\{\sigma_k\}$ is bounded away from one, then $\{\lambda_k\}$ is bounded away from one and hence, $\varphi(w_k) \to 0$; this is a contradiction.

(Case 2) On the other hand, for infinitely many k, if $\alpha_k < \bar{\alpha}_k$, we have that $\alpha_k \leq \theta\bar{\alpha}_k$. Then, condition (6.3) fails for an $\tilde{\alpha}_k$ with $\alpha_k < \tilde{\alpha}_k \leq \alpha_k/\theta = \theta^{t-1}\bar{\alpha}_k$. Notice that α_k/θ is the value corresponding to the last failure. Recall that the derivative of the real-valued function $\alpha \mapsto \phi(\alpha) := \varphi\left(\bar{\mathrm{Exp}}_{w_k}(\alpha\Delta w_k)\right)$ at some value α is

$$\phi'(\alpha) = \mathrm{D}\varphi\left(\bar{\mathrm{Exp}}_{w_k}(\alpha\Delta w_k)\right)\left[\mathrm{D}\bar{\mathrm{Exp}}_{w_k}(\alpha\Delta w_k)[\Delta w_k]\right]. \tag{6.32}$$

Applying the mean value theorem to $\phi(\alpha)$ on interval $[0, \tilde{\alpha}_k]$ yields a number $\xi \in (0,1)$ such that

$$\tilde{\alpha}_k\phi'(\xi\tilde{\alpha}_k) = \phi(\tilde{\alpha}_k) - \phi(0). \tag{6.33}$$

For short, let $u := \xi\tilde{\alpha}_k\Delta w_k$. Hence,

$$\begin{aligned}
\tilde{\alpha}_k\beta\langle \mathrm{grad}\,\varphi_k, \Delta w_k\rangle <& \phi(\tilde{\alpha}_k) - \phi(0) \text{ (since condition (6.3), i.e., (6.17), fails for } \tilde{\alpha}_k) \\
=& \tilde{\alpha}_k\phi'(\xi\tilde{\alpha}_k) \text{ (by (6.33))} \\
=& \tilde{\alpha}_k\mathrm{D}\varphi(\bar{\mathrm{Exp}}_{w_k}(u))\left[\mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right] \text{ (by (6.32))} \\
=& \tilde{\alpha}_k\left\langle \mathrm{grad}\,\varphi(\bar{\mathrm{Exp}}_{w_k}(u)), \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\rangle. \text{ (by Definition 19)}
\end{aligned} \tag{6.34}$$

On the other hand, note that

$$\begin{aligned}
\langle \mathrm{grad}\,\varphi_k, \Delta w_k\rangle &= \langle \mathrm{grad}\,\varphi_k, u\rangle / \xi\tilde{\alpha}_k \\
&= \left\langle \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\mathrm{grad}\,\varphi_k], \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[u]\right\rangle / \xi\tilde{\alpha}_k \text{ (by (4) of Proposition 4)} \\
&= \left\langle \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\mathrm{grad}\,\varphi_k], \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\rangle.
\end{aligned} \tag{6.35}$$

Subtracting $\tilde{\alpha}_k\langle \mathrm{grad}\,\varphi_k, \Delta w_k\rangle$ from both sides of (6.34) and using equalities (6.35) gives

$$\begin{aligned}
&\tilde{\alpha}_k(\beta-1)\langle \mathrm{grad}\,\varphi_k, \Delta w_k\rangle \\
<& \tilde{\alpha}_k\left[\left\langle \mathrm{grad}\,\varphi(\bar{\mathrm{Exp}}_{w_k}(u)), \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\rangle - \langle \mathrm{grad}\,\varphi_k, \Delta w_k\rangle\right] \\
=& \tilde{\alpha}_k\left\langle \mathrm{grad}\,\varphi(\bar{\mathrm{Exp}}_{w_k}(u)) - \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\mathrm{grad}\,\varphi_k], \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\rangle \\
\leq& \tilde{\alpha}_k\left\|\mathrm{grad}\,\varphi(\bar{\mathrm{Exp}}_{w_k}(u)) - \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\mathrm{grad}\,\varphi_k]\right\|\left\|\mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\| \\
=& \tilde{\alpha}_k\left\|\mathrm{grad}\,\varphi(y) - \mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\mathrm{grad}\,\varphi_k]\right\|\left\|\mathrm{D}\bar{\mathrm{Exp}}_{w_k}(u)[\Delta w_k]\right\| \text{ (by letting } y := \bar{\mathrm{Exp}}_{w_k}(u)) \\
\leq& \tilde{\alpha}_k\kappa\|u\|\|\Delta w_k\| \text{ (by Lipschitz continuity of } \varphi \text{ and (5) of Proposition 4)} \\
=& \kappa\xi\tilde{\alpha}_k^2\|\Delta w_k\|^2.
\end{aligned}$$

Finally, we obtain

$$\tilde{\alpha}_k(\beta-1)\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle < \kappa\xi\tilde{\alpha}_k^2\left\|\Delta w_k\right\|^2.$$

Consequently,

$$(\beta-1)\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle / (\kappa\xi\left\|\Delta w_k\right\|^2) < \tilde{\alpha}_k. \tag{6.36}$$

Because α_k satisfies condition (6.3) and $\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle < 0$, then

$$\begin{aligned}
\phi_k(0)-\phi_k(\alpha_k) \geq & -\alpha_k\beta\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle \\
\geq & -\theta\beta\tilde{\alpha}_k\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle \\
\geq & -\theta\beta\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle(\beta-1)\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle / (\kappa\xi\left\|\Delta w_k\right\|^2) \text{ (by (6.36))} \\
\geq & [\theta\beta(1-\beta)/\kappa\xi]\left(\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle / \left\|\Delta w_k\right\|\right)^2 \\
= & \omega\left(\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle / \left\|\Delta w_k\right\|\right),
\end{aligned}$$

where $\omega(\cdot)$ is an F-function (see [149, Definition 14.2.1 & 14.2.2 in P479]). Since $\{\varphi_k\}$ is bounded below and $\varphi_k \geq \varphi_{k+1}$, it follows that $\lim_{k\to\infty}(\varphi_k-\varphi_{k+1})=0$. By the definition of F-functions, we obtain

$$\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle / \left\|\Delta w_k\right\| \to 0.$$

Since $\{\|\Delta w_k\|\}$ is bounded (see Proposition 18), we have $\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle \to 0$. Choosing ρ_k with $z_k^T s_k/m \leq \rho_k \leq \|F(w_k)\|/\sqrt{m}$ implies that

$$\begin{aligned}
\left\langle \operatorname{grad}\varphi_k, \Delta w_k\right\rangle /(-2) = & \varphi_k - \sigma_k\rho_k z_k^T s_k \text{ (by Lemma 21)} \\
\geq & \varphi_k - \sigma_k\left\|F(w_k)\right\| z_k^T s_k/\sqrt{m} \\
\geq & \varphi_k - \sigma_k\left\|F(w_k)\right\|^2 \text{ (by (6.13))} \\
\geq & (1-\sigma_k)\varphi_k.
\end{aligned}$$

This shows that $\varphi(w_k) \to 0$, because $\{\sigma_k\}$ is bounded away from one; this is a contradiction. We complete the proof. □

6.8 Numerical Experiments

The numerical experiments compared the performance of the globally convergent RIPM (Algorithm 10) with those of other Riemannian methods. Here, we will test two problems: Nonnegative Low-Rank Matrix (NLRM) Approximation and Projection onto Nonnegative Stiefel Manifold, which were discussed in Section 1.3.2. They involve three manifolds:

$$\begin{aligned}
&\text{fixed-rank manifold} \quad \operatorname{Fr}(m,n,r) := \left\{X \in \mathbb{R}^{m\times n} : \operatorname{rank}(X) = r\right\}, \\
&\text{Stiefel manifold} \quad \operatorname{St}(n,k) := \left\{X \in \mathbb{R}^{n\times k} : X^T X = I_k\right\}, \\
&\text{Oblique manifold} \quad \operatorname{Ob}(n,k) := \left\{X \in \mathbb{R}^{n\times k} : \text{each column of } X \text{ has unit } l_2 \text{ norm}\right\}.
\end{aligned}$$

We only consider their embedded geometry and we apply the default retractions in Manopt, e.g., the retraction based on QR decomposition for the Stiefel manifold. Notice that although $\mathrm{Fr}(m,n,r)$ is not complete, RIPM is still valid in practice. The numerical experiments were performed in Matlab R2022a on a computer equipped with an Intel Core i7-10700 at 2.90GHz with 16GB of RAM. Our algorithms are built in the framework of Manopt 7.0 [37], a Riemannian optimization toolbox on Matlab. The code is freely available at https://github.com/GALVINLAI/RIPM.

6.8.1 Implementation Details

Parameters of Our RIPM

Our RIPM implementation (Algorithm 10) chooses the initial z_0 and s_0 from a uniform distribution in [0,1] and sets y_0 to zero if y exists. In Step 1, $\rho_k = z_k^T s_k/m$ and $\sigma_k = \min\{0.5, \|F(w_k)\|^{1/2}\}$. If the method is not specified in advance, we use the CR method to solve equation (5.32) and terminate it if the relative residual is smaller than 10^{-9} or the maximum number (1000) of iterates is reached. In Step 3, instead of finding the exact values of $\alpha_k^i, i = I, II$, we use a backtracking line search simultaneously for the central conditions and the sufficient decreasing condition. As a slight simplification (see [73]), we do not enforce the second central condition. Here, we set $\gamma_0 = 0.9, \gamma_{k+1} = (\gamma_k + 0.5)/2$; and $\beta = 10^{-4}$, $\theta = 0.5$.

Experimental Setting

We compared our method with the following Riemannian methods [128, 148]:

- RALM [128]: Riemannian augmented Lagrangian method.
- REPM$_{\text{lqh}}$ [128]: Riemannian exact penalty method with smoothing function of linear-quadratic and pseudo-Huber.
- REPM$_{\text{lse}}$ [128]: Riemannian exact penalty method with smoothing function of log-sum-exp.
- RSQP [148]: Riemannian sequential quadratic programming method.
- RIPM (Our method): Riemannian interior point method method (Algorithm 10).

Our experimental settings followed those of Obara et al. [148], where they used residuals based on the KKT conditions (5.3)-(5.7) to measure the deviation of an iterate from the set of KKT points. The KKT residual is defined by

$$\sqrt{\|\mathrm{grad}_x \mathcal{L}(w)\|^2 + \sum_{i=1}^{m}\{\min(0,z_i)^2 + \max(0,g_i(x))^2 + |z_i g_i(x)|^2\} + \sum_{j=1}^{l}|h_j(x)|^2 + \mathrm{Manvio}(x)},$$

where Manvio measures the violation of the manifold constraints. If $\mathcal{M} := \{x : f_j^{Man}(x) = 0, j = 1,\ldots,p\}$, then

$$\mathrm{Manvio}(x) := \sum_{j=1}^{p}|f_j^{Man}(x)|.$$

Only for the fixed-rank manifold $\mathrm{Fr}(m,n,r)$, we define $\mathrm{Manvio}(X) := 0$ if $\mathrm{rank}(X) = r$ and $+\infty$, otherwise. For the parameters of RALM, REPMs and RSQP, we utilize the experimental setting and Matlab codes provided by [148].

To measure the stability and speed of the algorithms, we conducted 20 random trials of each problem and model. In each trial, all the algorithms ran with the same initial point. The stopping criteria were based on the KKT residual, maximum iteration, maximum time, and changes in parameters. Concretely, each experiment terminated successfully if a solution with a KKT residual ε_{kkt} was found. For the *first-order* algorithms (including RALM and the REPMs), if the spent time exceeded $t_{\max}$ seconds, or the outer iteration number was over 1,000, or the algorithm did not update any parameters, we considered that the algorithm had terminated unsuccessfully. For the *second-order* algorithms (including RSQP, RIPM), if the spent time exceeded $t_{\max}$ seconds, or the iteration number was over 10,000, we considered that the algorithm had terminated unsuccessfully.

Here, considering that some problems might not have converged easily, the maximum number of iterations was chosen to be 1,000 (10,000), which was a sufficiently large value. The selection of $t_{\max}$ related to the actual time it took to run all the codes on the computer. Setting $t_{\max}$ too large resulted in excessive time spent on poorly performing algorithms. On the other hand, ε_{kkt} was chosen to better demonstrate that second-order algorithms could achieve more accurate solutions. Therefore, we chose the appropriate values for $t_{\max}$ and ε_{kkt} according to the problem that was to be solved.

6.8.2 Experiment I: Nonnegative Low-Rank Matrix Approximation

Problem Setting

The first experiment examines the problem of nonnegative low-rank matrix approximation in (NLRM). We will consider three cases $m = 20, 30, 40$ and let $n = 0.8m, r = 0.1m$. For each (m, n, r), we randomly generated nonnegative matrices $L \in \mathbb{R}^{m\times r}$ and $R \in \mathbb{R}^{r\times n}$ whose entries follow a uniform distribution in [0,1]. The original nonnegative matrix $A := LR$ is then obtained, and $\operatorname{rank}(A) = r$ with a very high probability. In the same way, we can generate a random feasible initial point x_0. Moreover, just like in [182], we add Gaussian noise with zero mean and different standard deviation $(\sigma = 0, 0.001, 0.01)$ to the original A. When there is no noise (i.e., $\sigma = 0$), the input data matrix A itself is exactly a solution.

Here, we set $t_{\max} = 180$ and $\varepsilon_{\text{kkt}} = 10^{-8}$. The tables of this subsection report the success rate (Rate) divided by the total number of trials, the average time in seconds (Time (s)), and the average iteration number (Iter.) among the successful trials. Boldface highlights the best results under the combined considerations of stability and speed; that is, for each setting, the Rate column shows in bold the success rates that are ≥ 0.95 and the Time column shows in bold the two (if any) fastest results among the algorithms with success rates ≥ 0.95.

Results and Analysis

The numerical results are shown in Table 6.1. RIPM performed the best. While the first-order algorithms (including RALM and the REPMs) were as fast but less stable. The time spent by RALM and the REPMs grew slowly with the problem size, but their success rates dropped sharply as the noise level (standard deviation σ) intensified, eventually leading to non-convergence.

In contrast, the convergence of the second-order algorithms (including RSQP, RIPM) was more stable, with RIPM being much faster than RSQP. The cost of a single iteration of RSQP drastically increased with the problem size. This is because RSQP requires solving a quadratic programming problem on the

Table 6.1 Performance of various Riemannian methods on problem (NLRM).

(m,n,r)	(20,16,2)			(30,24,3)			(40,32,4)		
no noise	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
RALM	0.4	1.115	31	0.65	1.813	31	0.75	2.800	31
REPM_{lqh}	**1**	**5.165**$\times 10^{-1}$	31	**1**	**1.009**	31	**1**	**1.747**	31
REPM_{lse}	**1**	2.242	31	**1**	4.041	31	**0.95**	6.952	31
RSQP	0.9	6.429	7	0.9	3.944$\times 10$	8	0.9	1.254$\times 10^{2}$	8
RIPM	**1**	**4.920**$\times 10^{-1}$	19	**1**	**2.247**	27	**1**	**5.277**	32
(m,n,r)	(20,16,2)			(30,24,3)			(40,32,4)		
$\sigma = 0.001$	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
RALM	0.2	1.001	31	0.15	2.050	31	0.05	2.758	31
REPM_{lqh}	0.1	4.983$\times 10^{-1}$	32	0.25	1.035	31	0.15	1.787	31
REPM_{lse}	0.15	2.444	31	0.1	4.867	31	0.05	8.371	31
RSQP	**0.95**	**6.619**	7	**0.95**	**3.848**$\times 10$	8	0.9	1.299$\times 10^{2}$	8
RIPM	**1**	**5.376**$\times 10^{-1}$	20	**1**	**2.342**	27	**1**	**4.631**	29
(m,n,r)	(20,16,2)			(30,24,3)			(40,32,4)		
$\sigma = 0.01$	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.	Rate	Time (s)	Iter.
RALM	0	-	-	0	-	-	0	-	-
REPM_{lqh}	0	-	-	0	-	-	0	-	-
REPM_{lse}	0	-	-	0	-	-	0	-	-
RSQP	**1**	**7.295**	8	**0.95**	**4.114**$\times 10$	8	**0.95**	**1.430**$\times 10^{2}$	9
RIPM	**1**	**5.980**$\times 10^{-1}$	21	**0.95**	**1.883**	25	**0.95**	**4.602**	29

tangent space of x_k in each iteration. As with RIPM, there is no explicit matrix form available. RSQP transforms it into a matrix representation form (similar to Step 1-6 in Algorithm 7) before using a quadratic programming solver. Instead, our RIPM avoids the expensive computation of the matrix representation by using the Krylov subspace methods. As can be seen from the table, RIPM takes the same amount (order of magnitude) of time as RALM and the REPMs.

6.8.3 Experiment II: Projection onto Nonnegative Stiefel Manifold

Problem Setting

The second experiment examines the problem of projecting onto the nonnegative part of Stiefel manifold in (Model_St), and its equivalent formulation (Model_Ob). Note that the former is a problem on Stiefel manifold, while the latter is on Oblique manifold. We examine both models on different manifolds.

We will consider the cases of $n = 40, 50, 60, 70$ and let $k = 0.2n$. For a general matrix C, it is always difficult to seek nonnegative projections globally. Fortunately, [116, Proposition 1] showed a way to construct matrix C such that (Model_St) has a unique and known solution X^*. First, we generate a random feasible point B of (Model_St); then, we obtained C by using the following Matlab code:

```
X1=(B>0).*(1+rand(n,k)); Xstar =X1./sqrt(sum(X1.*X1));
L=rand(k,k); L=L+k*eye(k); C=Xstar*L'.
```

The initial point is computed by projecting C onto the Stiefel manifold, that is obtained by

```
[U,~,V]=svd(C,'econ'); X0=U*V'.
```

The same settings are applied to (Model_Ob), except for

```
p=1; V=ones(k,p); V=V/norm(V,"fro").
```

We set $t_{\max} = 600$ and $\varepsilon_{\text{kkt}} = 10^{-6}$ for both models (Model_St) and (Model_Ob). Since the true solution is known, we added an Error column showing the average error $\|X_{\text{final}} - X^*\|_{\text{F}}$. Here, X_{final} denotes the final iterate point of a successful trial. As a full demonstration, in the second model (Model_Ob), we also tested RIPM by using the matrix representation method (Algorithm 7) to solve the Newton equation; this is denoted as RIPM_RepMat.

The tables of this subsection report the success rate (Rate) divided by the total number of trials, the average time in seconds (Time (s)), and the average iteration number (Iter.) among the successful trials. Boldface highlights the best results under the combined considerations of stability and speed; that is, for each setting, the Rate column shows in bold the success rates that are ≥ 0.95 and the Time column shows in bold the two (if any) fastest results among the algorithms with success rates ≥ 0.95.

Results and Analysis

The numerical results are listed in Table 6.2 and 6.3. The Error columns show that if the KKT residual is sufficiently small, then X_{final} does approximate the true solution. In particular, the second-order algorithms (including RSQP, RIPM) yield a more accurate solution (the error is less than 10^{-7}). From Table 6.2, we can see that RALM is stable and fast for (Model_St). However, from Table 6.3, RALM's success rate of convergence for (Model_Ob) decreases as the problem size becomes larger. The REPMs do not work at all on either model. RSQP also does not perform well on both models. RIPM_RepMat and RIPM successfully solved all instances of both models, though the time taken by RIPM_RepMat grew explosively. Overall, RIPM using the iterative method was fast and the most stable.

6.9 Summary

In the previous chapter, under standard assumptions, we established the local convergence of the Riemannian Interior Point Method (RIPM). This chapter continues to investigate and proves the global convergence of the RIPM when combined with classical linear search and the merit function $\varphi(w) = \|F(w)\|^2$. Numerical experiments demonstrate the stability and efficiency of our method.

Table 6.2 Performance of various Riemannian methods on (Model_St).

(n,k)	(40,8)				(50,10)			
	Rate	Time (s)	Iter.	Error	Rate	Time (s)	Iter.	Error
RALM	**1**	**2.347**	45	5.41×10^{-7}	**1**	**4.344**	54	5.21×10^{-7}
REPM_{lqh}	0	-	-	-	0	-	-	-
REPM_{lse}	0	-	-	-	0	-	-	-
RSQP	0.9	1.352×10	7	2.05×10^{-9}	0.7	3.097×10	6	2.47×10^{-9}
RIPM	**1**	**2.225**	31	3.72×10^{-8}	**1**	**3.785**	32	3.38×10^{-8}
(n,k)	(60,12)				(70,14)			
	Rate	Time (s)	Iter.	Error	Rate	Time (s)	Iter.	Error
RALM	**1**	**4.097**	34	4.93×10^{-7}	**1**	**6.234**	37	5.34×10^{-7}
REPM_{lqh}	0	-	-	-	0	-	-	-
REPM_{lse}	0	-	-	-	0	-	-	-
RSQP	0.65	7.802×10	7	6.48×10^{-9}	0.85	1.661×10^{2}	7	2.64×10^{-9}
RIPM	**1**	**5.555**	32	2.81×10^{-8}	**1**	**7.574**	33	2.45×10^{-8}

Table 6.3 Performance of various Riemannian methods on (Model_Ob).

(n,k)	(40,8)				(50,10)			
	Rate	Time (s)	Iter.	Error	Rate	Time (s)	Iter.	Error
RALM	**1**	**2.510**	51	5.04×10^{-7}	**0.95**	**4.727**	64	4.94×10^{-7}
REPM_{lqh}	0	-	-	-	0	-	-	-
REPM_{lse}	0	-	-	-	0	-	-	-
RSQP	0.65	8.618	5	2.30×10^{-10}	0.7	2.782×10	6	1.12×10^{-10}
RIPM	**1**	**3.791**	22	5.62×10^{-9}	**1**	**5.880**	23	7.93×10^{-9}
RIPM_RepMat	**1**	1.954×10	31	4.34×10^{-8}	**1**	4.718×10	32	3.56×10^{-8}
(n,k)	(60,12)				(70,14)			
	Rate	Time (s)	Iter.	Error	Rate	Time (s)	Iter.	Error
RALM	0.6	5.725	49	3.82×10^{-7}	0.6	8.223	52	3.85×10^{-7}
REPM_{lqh}	0	-	-	-	0	-	-	-
REPM_{lse}	0	-	-	-	0	-	-	-
RSQP	0.7	4.446×10	5	1.17×10^{-9}	0.5	9.138×10	5	1.82×10^{-9}
RIPM	**1**	**7.134**	23	9.69×10^{-9}	**1**	**9.268**	24	1.06×10^{-8}
RIPM_RepMat	**1**	**1.018**$\times10^{2}$	32	3.20×10^{-8}	**1**	**1.861**$\times10^{2}$	33	2.75×10^{-8}

Chapter 7

Several Theoretical Results for Quasi-Newton RIPM

Part	Section
Part 1. Preparation work	7.1 Quasi-Newton RIPM
Part 2. Theoretical results	7.2 Local and Linear Convergence of Quasi-Newton RIPM 7.3 Local and Superlinear Convergence of Quasi-Newton RIPM
Part 3. Proofs	7.4 Collection of Proofs

In this chapter, we study the several theoretical results about local convergence of quasi-Newton Riemannian Interior Point Methods (RIPM). The quasi-Newton RIPM can approximate the Hessian of Lagrangian in (5.9) with gradient information while ensuring its local convergence. First of all, Section 7.1 introduces the classical quasi-Newton method and then quasi-Newton RIPM algorithm. In Section 7.2, we state a sufficient condition, called *bounded deterioration property*, for the locally linear convergence of quasi-Newton RIPM. In Section 7.3, we establish the locally superlinear convergence of quasi-Newton RIPM by using an analogous *Dennis Moré condition*. For convenience, we have put the technical proofs all together in Section 7.4. However, there is still a great deal of work to be done to refine the quasi-Newton RIPM. As the last chapter of this thesis, we only give some constructive results.

7.1 Quasi-Newton RIPM

In classical Euclidean optimization, the interior point method is a typical "second-order" optimization algorithm because it requires computing the Hessian matrices at the current point at every iteration, i.e., it requires obtaining the second-order information of the functions based on that point (note that it is necessary to reacquire it at every iteration). In the case of simple problems such as linear programming and quadratic programming, where the objective and constraint functions are linear or quadratic, computing their Hessian matrices is trivial. However, for general nonconvex nonlinear objective and constraint functions in (CEO), there are also some difficulties in obtaining their Hessian matrices, as follows:

1. General nonconvex nonlinear functions are too complex (e.g., training of deep neural networks), so their Hessian matrices may be intractable, or it is mathematically difficult to find their closed

form. In addition, In addition, if the user has to manually provide the solver with second-order information about the optimization problem, it raises the bar for the user, which is not conducive to the popularization of the algorithm.

2. Even if the Hessian matrix has closed from, the computational cost is unacceptable in practice. This will cause it to be a bottleneck in the practice of interior point method.

To overcome the above issues, the interior point method in the Euclidean setting borrows the idea of quasi-Newton methods — approximating Hessian matrices using gradients, i.e., using only first-order information. This greatly reduces the computational effort. Below we briefly review the whole process of quasi-Newton method for unconstrained minimization in the Euclidean setting, see Algorithm 11.

Algorithm 11: Quasi-Newton Method for (UEO)

Input: An objective function f defined on $\mathbb{R}^n$, an initial point $x_0 \in \mathbb{R}^n$, and an initial Hessian approximation $H_0 \in \mathbb{R}^{n\times n}$.

Output: Sequence $\{x_k\} \subset \mathbb{R}^n$ that converges to the minimizer of f.

Set $k \to 0$;

while *stopping criterion not satisfied* **do**

1. Compute the gradient $\operatorname{egrad} f(x_k) \in \mathbb{R}^n$;
2. Compute the quasi-Newton direction by solving linear equation $H_k d_k = -\operatorname{egrad} f(x_k)$;
3. Compute a step size $t_k > 0$;
4. Compute the next point as $x_{k+1} := x_k + t_k d_k$;
5. Compute H_{k+1} using an update formula (e.g., BFGS update);
6. $k \to k + 1$;

end

In Step 5 above, different update rules exist for H_{k+1}. The most popular one is called Broyden-Fletcher-Goldfarb-Shanno (BFGS) update, as described below:

$$H_{k+1} := H_k + \frac{yy^T}{y^T s} - \frac{H_k s s^T H_k}{s^T H_k s}$$

where

$$s = x_{k+1} - x_k, \quad y = \operatorname{egrad} f(x_{k+1}) - \operatorname{egrad} f(x_k).$$

There are also DFP, SR1, Broyden (family), etc., see [59, 119, 147] for details. Regardless of which update formula is used, as long as $H_k \in \mathbb{R}^{n\times n}$ satisfies certain conditions (i.e., expressing how H_k approximates the original Hessian $\mathbf{H}_f(x) \in \mathbb{R}^{n\times n}$) then Algorithm 11 will have the same locally convergence as the standard Newton method. Using the same idea, in the interior point method we can also construct Hessians using gradients. Specifically, i.e., the matrix $\mathbf{H}_{\mathcal{L}}(w_k)$ in (5.22) (in fact, this is the only place in the interior point method where second-order information is needed) is replaced by an approximation matrix G_k; all others remain the same. This is the *quasi-Newton RIPM* in Euclidean setting and can be found in papers [134, 208, 205].

Next, it is natural to apply on this thought to Riemannian setting. Let us consider the prototype Algorithm 6 of RIPM. We define an approximate operator of $\nabla F(w_k)\colon T_{w_k}\mathcal{N} \to T_{w_k}\mathcal{N}$, denoted by $B_k\colon T_{w_k}\mathcal{N} \to T_{w_k}\mathcal{N}$, by replacing $\operatorname{Hess}_x \mathcal{L}(w_k)$ in the first row of (5.17) with some $G_k\colon T_{x_k}\mathcal{M} \to$

$T_{x_k}\mathcal{M}$ and keeping everything else the same:

$$B_k\Delta w := \begin{pmatrix} G_k\Delta x + \sum_{j=1}^{l}\Delta y_j \operatorname{grad} h_j(x_k) + \sum_{i=1}^{m}\Delta z_i \operatorname{grad} g_i(x_k) \\ \langle \operatorname{grad} h_j(x_k), \Delta x\rangle, \text{ for } j = 1, 2, \ldots, l \\ \langle \operatorname{grad} g_i(x_k), \Delta x\rangle + \Delta s_i, \text{ for } i = 1, 2, \ldots, m \\ Z_k\Delta s + S_k\Delta z \end{pmatrix} \tag{7.1}$$

for all $\Delta w = (\Delta x, \Delta y, \Delta s, \Delta z) \in T_{w_k}\mathcal{N}$. Now, we obtain quasi-Newton RIPM by replacing (5.28) in Algorithm 6 with $B_k\Delta w_k = -F(w_k) + \mu_k\hat{e}$. For the sake of completeness, we formulate the algorithm as follows, where the simple step scheme (5.29) is chosen.

Algorithm 12: Prototype Algorithm of Quasi-Newton RIPM for (CRO)

Input: A problem of (CRO), an initial point $w_0 = (x_0, y_0, z_0, s_0) \in \mathcal{N}$ with $(z_0, s_0) > 0$ and a retraction R on $\mathcal{M}$.

Output: Sequence $\{x_k\} \subset \mathcal{M}$.

Set $k \to 0$, $\mu_0 > 0$, $0 < \hat{\gamma} \leq 1$;

while *stopping criterion not satisfied* **do**

1. Solve the perturbed Newton equation (a linear operator equation on tangent space $T_{w_k}\mathcal{N}$):

$$B_k\Delta w_k = -F(w_k) + \mu_k\hat{e} \tag{7.2}$$

to obtain $\Delta w_k = (\Delta x_k, \Delta y_k, \Delta z_k, \Delta s_k) \in T_{w_k}\mathcal{N}$;

2. Choose $\hat{\gamma} \leq \gamma_k \leq 1$ and compute the step size:

$$\alpha_k := \min\left\{1, \gamma_k \min_i\left\{-\frac{(s_k)_i}{(\Delta s_k)_i} \mid (\Delta s_k)_i < 0\right\}, \gamma_k \min_i\left\{-\frac{(z_k)_i}{(\Delta z_k)_i} \mid (\Delta z_k)_i < 0\right\}\right\}.$$

3. Compute the next point as $w_{k+1} = (x_{k+1}, y_{k+1}, z_{k+1}, s_{k+1}) := \bar{\mathrm{R}}_{w_k}(\alpha_k\Delta w_k)$;
4. Choose $0 < \mu_{k+1} < \mu_k$;
5. $k \to k + 1$;

end

The operator G_k should of course be properly chosen to approximate the original Hessian operator $\mathrm{Hess}_x\,\mathcal{L}(w_k)$. In following sections, we will describe the properties that characterize G_k such that we can obtain the same local convergence as Algorithm 6. To distinguish, we will refer to Algorithm 6 as the *Newton RIPM*.

7.2 Local and Linear Convergence of Quasi-Newton RIPM

In this section, we will show local and linear convergence of quasi-Newton RIPM. Throughout this chapter, let $\mathcal{M}$ be a Riemannian manifold endowed with an isometric vector transport T and an associated retraction R. Let $w^* = (x^*, y^*, z^*, s^*)$ be a solution point of (CRO) and the current point $w_k = (x_k, y_k, z_k, s_k)$

sufficiently close to w^*. We define

$$\zeta_k^x := \mathrm{R}_{x^*}^{-1}(x_k), \zeta_k^y := y_k - y^*, \zeta_k^z := z_k - z^*, \zeta_k^s := s_k - s^*,$$

then, by (2.43) we have

$$w_k = \bar{\mathrm{R}}_{w^*}(\zeta_k) = \left(\mathrm{R}_{x^*}(\zeta_k^x), y^* + \zeta_k^x, z^* + \zeta_k^z, s^* + \zeta_k^s\right),$$

where $\zeta_k := (\zeta_k^x, \zeta_k^y, \zeta_k^z, \zeta_k^s) \in T_{w^*}\mathcal{N}$. The following lemma is an auxiliary result for Theorem 21 and is a generalization of [208, Lemma 7].

Lemma 25. *Consider the Algorithm 12 for solving problem (CRO). Suppose that (A1)-(A4) hold at some w^* and Algorithm 12 generates the sequence $\{w_k\}$. If there exist positive constants ε, δ such that for all w_k and linear operators $G_k \colon T_{x_k}\mathcal{M} \to T_{x_k}\mathcal{M}$ satisfying*

$$d(w_k, w^*) \le \varepsilon, \text{ and } \left\| G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right\| \le \delta, \tag{7.3}$$

where $\zeta_k^x = \mathrm{R}_{x^*}^{-1}(x_k) \in T_{x^*}\mathcal{M}$, then the following hold:

(i) there exist positive constants c_1, c_2 such that for all $k \ge 0$,

$$\left\| B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1} \right\| \le \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon},$$

where $\zeta_k = \bar{\mathrm{R}}_{w^*}^{-1}(w_k) \in T_{w^*}\mathcal{N}$;

(ii) there exists a positive constant Φ such that for all $k \ge 0$, B_k is nonsingular and $\|B_k^{-1}\| \le \Phi$;

(iii) furthermore, if we choose the parameter μ_k such that $\mu_k = O(\|F(w_k)\|)$, we have

$$1 - \alpha_k \le (1 - \gamma_k) + O(F(w_k)) + O(\mu_k). \tag{7.4}$$

The bounded deterioration property [39, Theorem 3.2] is a well-known sufficient condition for the general local convergence of quasi-Newton methods. It covers most of the quasi-Newton update formulas, such as BFGS. The next theorem shows the local convergence property of our quasi-Newton RIPM. We require the sequence of linear operators $\{G_k\}$ on $T_{x_k}\mathcal{M}$ to satisfy an analogous bounded deterioration property. Theorem 21 is a generalization of [208, Theorem 2] to a Riemannian manifold.

Theorem 21 (Local and Linear Convergence of Quasi-Newton RIPM)**.** *Consider the Algorithm 12 for solving problem (CRO) and choose the parameter μ_k such that $\mu_k = O(\|F(w_k)\|^{1+\tau})$ for some positive constant τ. Let (A1)-(A4) hold at some w^*. Suppose further that the sequence of linear operators $\{G_k\}$ with $G_k \colon T_{x_k}\mathcal{M} \to T_{x_k}\mathcal{M}$ satisfies the* bounded deterioration property:

$$\left\| G_{k+1} - \mathrm{T}_{\zeta_{k+1}^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_{k+1}^x}^{-1} \right\| \le (1 + \beta_1 \sigma_k) \left\| G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right\| + \beta_2 \sigma_k, \tag{7.5}$$

where $\zeta_k^x = \mathrm{R}_{x^*}^{-1}(x_k)$, β_1 and β_2 are some positive constants, and $\sigma_k := \max\{d(w_k, w^*), d(w_{k+1}, w^*)\}$.

Then for Algorithm 12 and any $\nu \in (1-\hat{\gamma}, 1)$, there exist positive constants $\varepsilon \equiv \varepsilon(\nu)$ and $\delta \equiv \delta(\nu)$ such that, for all $w_0 \in \mathcal{N}$ and initial operator G_0 with

$$d(w_0, w^*) \leq \varepsilon, \text{ and } \left\| G_0 - \mathrm{T}_{\zeta_0^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_0^x}^{-1} \right\| \leq \frac{1}{2}\delta,$$

the sequence $\{w_k\}$ is well-defined and converges to w^*. Furthermore, for each $k \geq 0$, we have

$$\frac{d(w_{k+1}, w^*)}{d(w_k, w^*)} \leq \nu, \text{ and } \left\| G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right\| \leq \delta. \tag{7.6}$$

7.3 Local and Superlinear Convergence of Quasi-Newton RIPM

The well-known Dennis Moré condition [58, Theorem 2.2] has a very important place in the analyses of quasi-Newton methods in the Euclidean setting. It is a necessary and sufficient condition for superlinear convergence of the quasi-Newton method. This condition encompasses a large category of well-known update formulas, such as BFGS, DFP. Gallivan et al. [81] have generalized it to the Riemannian quasi-Newton methods. In this section, we give an analogous Dennis Moré condition for our quasi-Newton RIPM in Theorem 22 and establish its superlinear convergence in Theorem 23. On the basis of the previous section, we assume that a well-defined sequence $\{w_k\}$ converges linearly to w^*.

Theorem 22 (Dennis Moré Condition for Quasi-Newton RIPM)**.** *Consider the Algorithm 12 for solving problem (CRO). Suppose that (A1)-(A4) hold at some w^*; the sequence $\{w_k\}$ generated by Algorithm 12 converges linearly to w^*; and the sequence $\{\|B_k^{-1}\|\}$ is bounded. Choose the parameters μ_k, γ_k such that*

$$\mu_k = o(\|F(w_k)\|) \text{ and } \gamma_k \to 1. \tag{7.7}$$

Then, the following are equivalent:

(a) the sequence of linear operators $\{B_k\}$ satisfies

$$\lim_{k\to\infty} \frac{\left\| \left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1} \right) \Delta w_k \right\|}{\|\Delta w_k\|} = 0, \tag{7.8}$$

where $\zeta_k = \bar{\mathrm{R}}_{w^*}^{-1}(w_k) \in T_{w^*}\mathcal{N}$;

(b) the sequence $\{F(w_k)\}$ satisfies

$$\lim_{k\to\infty} \frac{\left\| \bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) \right\|}{\|\alpha_k \Delta w_k\|} = 0;$$

(c) the sequence $\{F(w_k)\}$ satisfies

$$\lim_{k\to\infty} \frac{\|F(w_{k+1})\|}{\|\Delta w_k\|} = 0;$$

(d) the sequence $\{w_k\}$ converges superlinearly to w^*, i.e.,

$$\lim_{k\to\infty} \frac{d(w_{k+1}, w^*)}{d(w_k, w^*)} = 0.$$

For the superlinear convergence, (a) of Theorem 22 requires the property for $\{B_k\}$. However, next theorem can guarantee the convergence by a direct requirement for $\{G_k\}$. Please compare (7.8) and (7.9).

Theorem 23 (Superlinear Convergence of Quasi-Newton RIPM)**.** *Consider the Algorithm 12 for solving problem (CRO). Suppose that (A1)-(A4) hold at some w^*. Choose the parameters μ_k, γ_k such that $\mu_k = o(\|F(w_k)\|)$ and $\gamma_k \to 1$. If the sequence of linear operators $\{G_k\}$ with $G_k : T_{x_k}\mathcal{M} \to T_{x_k}\mathcal{M}$ satisfies the bounded deterioration property (7.5) and*

$$\lim_{k\to\infty} \frac{\left\| \left(G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right) \Delta x_k \right\|}{\|\Delta x_k\|} = 0, \tag{7.9}$$

where $\zeta_k^x = \mathrm{R}_{x^*}^{-1}(x_k)$, then the sequence $\{w_k\}$ generated by Algorithm 12 converges locally and superlinearly to w^*.

7.4 Collection of Proofs

7.4.1 Proof of Lemma 25

Proof. **(i)** Take any $\Delta w = (\Delta x, \Delta y, \Delta s, \Delta z) \in T_{w_k}\mathcal{N}$. First, we compute

$$\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1} \right) \Delta w.$$

Using the vector transport on the product manifold $\bar{\mathrm{T}}_{\zeta_k}$ in (2.45), its inverse $\bar{\mathrm{T}}_{\zeta_k}^{-1}$ in (2.46), and $\nabla F(w^*)$ in (5.17), we obtain that $\bar{\mathrm{T}}_{\zeta_k} \nabla F(w) \bar{\mathrm{T}}_{\zeta_k}^{-1} \Delta w$ is equal to

$$\begin{pmatrix} \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \Delta x + \sum_{j=1}^{l} (\Delta y)_j \mathrm{T}_{\zeta_k^x} \operatorname{grad} h_j(x^*) + \sum_{i=1}^{m} (\Delta z)_i \mathrm{T}_{\zeta_k^x} \operatorname{grad} g_i(x^*) \\ \langle \operatorname{grad} h_j(x^*), \mathrm{T}_{\zeta_k^x}^{-1} \Delta x \rangle, \text{ for } j = 1, 2, \ldots, l \\ \langle \operatorname{grad} g_i(x^*), \mathrm{T}_{\zeta_k^x}^{-1} \Delta x \rangle + (\Delta s)_i, \text{ for } i = 1, 2, \ldots, m \\ Z^* \Delta s + S^* \Delta z \end{pmatrix}. \tag{7.10}$$

Then, subtracting (7.10) from $B_k \Delta w$ in (7.1) yields $\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1} \right) \Delta w$, which is equal to

$$\begin{pmatrix} \left(G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right) \Delta x + \sum_{j=1}^{l} (\Delta y)_j [\operatorname{grad} h_j(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grad} h_j(x^*)] \\ \qquad + \sum_{i=1}^{m} (\Delta z)_i [\operatorname{grad} g_i(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grad} g_i(x^*)] \\ \langle \operatorname{grad} h_j(x_k), \Delta x \rangle - \langle \operatorname{grad} h_j(x^*), \mathrm{T}_{\zeta_k^x}^{-1} \Delta x \rangle, \text{ for } j = 1, 2, \ldots, l \\ \langle \operatorname{grad} g_i(x_k), \Delta x \rangle - \langle \operatorname{grad} g_i(x^*), \mathrm{T}_{\zeta_k^x}^{-1} \Delta x \rangle, \text{ for } i = 1, 2, \ldots, m \\ (Z_k - Z^*) \Delta s + (S_k - S^*) \Delta z \end{pmatrix}. \tag{7.11}$$

Consider h_j, for $j = 1, 2, \ldots, l$. By the isometry of vector transport T, we get

$$\langle \operatorname{grad} h_j(x^*), \mathrm{T}_{\zeta_k^x}^{-1} \Delta x \rangle = \langle (\mathrm{T}_{\zeta_k^x}^{-1})^* \operatorname{grad} h_j(x^*), \Delta x \rangle = \langle \mathrm{T}_{\zeta_k^x} \operatorname{grad} h_j(x^*), \Delta x \rangle.$$

Hence, the second row of (7.11) can be simplified as $\langle \operatorname{grad} h_j(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grad} h_j(x^*), \Delta x \rangle$. In the same way, consider g_i, for $i = 1, \ldots, m$, the third row of (7.11) is equal to $\langle \operatorname{grad} g_i(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grad} g_i(x^*), \Delta x \rangle$.

For simplification, we introduce the following notations:

$$\begin{aligned}
\theta_j &:= \operatorname{grad} h_j(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grad} h_j(x^*), \text{ for } j = 1, 2, \ldots, l, \\
\lambda_i &:= \operatorname{grad} g_i(x_k) - \mathrm{T}_{\zeta_k^x} \operatorname{grod} g_i(x^*), \text{ for } i = 1, 2, \ldots, m, \\
\alpha_1 &:= \left(G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1}\right) \Delta x.
\end{aligned}$$

Then, (7.11) reduces to

$$\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1}\right) \Delta w = \begin{pmatrix} \alpha_1 + \sum_{j=1}^{l} (\Delta y)_j \theta_j + \sum_{i=1}^{m} (\Delta z)_i \lambda_i \\ \langle \theta_j, \Delta x \rangle, \text{ for } j = 1, 2, \ldots, l \\ \langle \lambda_i, \Delta x \rangle, \text{ for } i = 1, 2, \ldots, m \\ (Z_k - Z^*)\Delta s + (S_k - S^*)\Delta z \end{pmatrix}. \tag{7.12}$$

Note that, smoothness and Lipschitz continuity of $\{h_j\}, \{g_i\}$ at x^* (see Definition 29) imply that

$$\|\theta_j\| \leq \kappa \|\zeta_k^x\| \text{ for } j = 1, 2, \ldots, l, \text{ and } \|\lambda_i\| \leq \kappa \|\zeta_k^x\| \text{ for } i = 1, 2, \ldots, m. \tag{7.13}$$

Without loss of generality, we can take a single constant κ for each h_j, g_i above. Again, we introduce the following notations: $\alpha := \alpha_1 + \alpha_2$, and $\alpha_2 := \sum_{j=1}^{l} (\Delta y)_j \theta_j + \sum_{i=1}^{m} (\Delta z)_i \lambda_i$, $\beta_j := \langle \theta_j, \Delta x \rangle$ for $\beta \in \mathbb{R}^l$, $\gamma_i := \langle \lambda_i, \Delta x \rangle$ for $\gamma \in \mathbb{R}^m$, $\delta := \delta_1 + \delta_2 \in \mathbb{R}^m$, and $\delta_1 := (Z_k - Z^*)\Delta s$, and $\delta_2 := (S_k - S^*)\Delta z$. These notations give $\|(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1}) \Delta w\|^2 = \|\alpha\|_{x_k}^2 + \|\beta\|^2 + \|\gamma\|^2 + \|\delta\|^2$.

Now, let us examine each term on the right-hand side of the equation above. First, we assert that

$$\begin{aligned}
\|\beta\|^2 =& \sum_{j}^{l} |\langle \theta_j, \Delta x \rangle|^2 \leq \sum_{j}^{l} (\|\theta_j\| \|\Delta x\|)^2 \text{ (by the Cauchy–Schwarz inequality)} \\
\leq& l\kappa^2 \|\zeta_k^x\|^2 \|\Delta x\|^2 \text{ (by (7.13))} \\
\leq& l\kappa^2 \frac{1}{a_0^2} d^2(w_k, w^*) \|\Delta w\|^2 \text{ (by (iv) of Lemma 6, (2.41) and (2.42))} \\
=& O(d^2(w_k, w^*)) \|\Delta w\|^2.
\end{aligned} \tag{7.14}$$

In a similar manner, we obtain $\|\gamma\|^2 = O(d^2(w_k, w^*)) \|\Delta w\|^2$. Next, for $\|\delta\|^2$ we see that

$$\begin{aligned}
\|\delta_1\| =& \|(Z_k - Z^*)\Delta s\| \leq \|Z_k - Z^*\|_{\mathrm{F}} \|\Delta s\| \text{ (by the properties of the Frobenius norm)} \\
=& \|z_k - z^*\| \|\Delta s\| \text{ (since } Z_k, Z^* \text{ are diagonal matrices)} \\
\leq& d(w_k, w^*) \|\Delta w\| \text{ (by (2.41) and (2.42)).}
\end{aligned}$$

Similarly, $\|\delta_2\| \leq d(w_k, w^*)\,\|\Delta w\|$. Thus, $\|\delta\|^2 = O(d^2(w_k, w^*))\,\|\Delta w\|^2$. To deal with $\|\alpha\|^2$, we note that $\|\alpha\|^2 = \|\alpha_1 + \alpha_2\|^2 \leq (\|\alpha_1\| + \|\alpha_2\|)^2 = \|\alpha_1\|^2 + \|\alpha_2\|^2 + 2\|\alpha_1\|\|\alpha_2\|$. Next, consider

$$\begin{aligned}
\|\alpha_2\| =& \|\sum_j^l (\Delta y)_j \theta_j + \sum_i^m (\Delta z)_i \lambda_i\| \leq \sum_j^l |\,(\Delta y)_j\,|\,\|\theta_j\| + \sum_i^m |(\Delta z)_i|\,\|\lambda_i\| \\
\leq& \kappa\,\|\zeta_k^x\| \sum_j^l |\,(\Delta y)_j\,| + \kappa\,\|\zeta_k^x\| \sum_i^m |(\Delta z)_i| \ \text{(by (7.13))} \\
=& \kappa\,\|\zeta_k^x\|\,(\|\Delta y\|_1 + \|\Delta z\|_1) \ \text{(by the definition of the } l_1 \text{ norm)} \\
\leq& \kappa\,\|\zeta_k^x\|\,[\sqrt{l}\,\|\Delta y\| + \sqrt{m}\,\|\Delta z\|] \ \text{(from the relationship of the } l_1 \text{ and } l_2 \text{ norms)} \\
\leq& \kappa\,\|\zeta_k^x\|\,[\sqrt{l}\,\|\Delta w\| + \sqrt{m}\,\|\Delta w\|] \ \text{(by (2.41))} \\
\leq& \kappa \max\{\sqrt{l}, \sqrt{m}\}\,\|\zeta_k^x\|\,\|\Delta w\| \\
\leq& \kappa \max\{\sqrt{l}, \sqrt{m}\} d(x_k, x^*)\,\|\Delta w\|\,/a_0 \ \text{(by (iv) of Lemma 6)} \\
\leq& \kappa \max\{\sqrt{l}, \sqrt{m}\} d(w_k, w^*)\,\|\Delta w\|\,/a_0 \ \text{(by (2.42))} \\
=& O(d(w_k, w^*))\,\|\Delta w\|\,. \qquad (7.15)
\end{aligned}$$

Finally, combining the results from (7.12) to (7.15), we conclude that

$$\begin{aligned}
&\left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F\,(w^*)\,\bar{\mathrm{T}}_{\zeta_k}^{-1}\right)\Delta w\right\|^2 \\
\leq& \left\|\left(G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x\,\mathcal{L}\,(w^*)\,\mathrm{T}_{\zeta_k^x}^{-1}\right)\Delta x\right\|^2 + O(d^2(w_k, w^*))\,\|\Delta w\|^2 \\
&+ O(d(w_k, w^*))\,\|\Delta w\|\,\left\|\left(G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x\,\mathcal{L}\,(w^*)\,\mathrm{T}_{\zeta_k^x}^{-1}\right)\Delta x\right\|. \qquad (7.16)
\end{aligned}$$

Define operators

$$S := B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F\,(w^*)\,\bar{\mathrm{T}}_{\zeta_k}^{-1},\ T := G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x\,\mathcal{L}\,(w^*)\,\mathrm{T}_{\zeta_k^x}^{-1}.$$

Now, take $\Delta w = (\Delta x, \Delta y, \Delta s, \Delta z) \in T_{w_k}\mathcal{N}$ such that $\|\Delta w\| = 1$. From (7.3), we have $d(w_k, w^*) \leq \varepsilon$; thus, inequality (7.16) becomes

$$\|S\Delta w\|^2 \leq \|T\Delta x\|^2 + \|T\Delta x\| c_1\varepsilon + c_2\varepsilon \qquad (7.17)$$

for some constant $c_1, c_2 > 0$. Consider Δx, which is a component of Δw of the unit norm; (2.41) implies $\|\Delta x\| \leq \|\Delta w\| = 1$. Thus, we have

$$\begin{aligned}
\|T\Delta x\| &\leq \sup\{\|T\tilde{\Delta} x\| \mid \|\tilde{\Delta} x\| \leq 1, \tilde{\Delta} x \in T_{x_k}\mathcal{M}\} \ \text{(since } \|\Delta x\| \leq 1\text{)} \\
&= \|T\| \ \text{(by the definition of operator norm)} \\
&\leq \delta \ \text{(by assumption (7.3)).} \qquad (7.18)
\end{aligned}$$

Finally, we have $\|S\| = \sup\{\|S\Delta w\| \mid \|\Delta w\| = 1, \Delta w \in T_{w_k}\mathcal{N}\} \leq \sqrt{\delta^2 + \delta c_1\varepsilon + c_2\varepsilon}$. Thus, this proves assertion (i).

(ii) Since $\bar{\mathrm{T}}$ is isometric, $\|(\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1})^{-1}\| = \|\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1}\| = \|\nabla F\left(w^*\right)^{-1}\|$. By choosing ε and δ such that

$$\sqrt{\delta^2+\delta c_1\varepsilon+c_2\varepsilon}\|\nabla F\left(w^*\right)^{-1}\|<1,$$

we have

$$\|(\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1})^{-1}\|\|B_k-\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\| \le \sqrt{\delta^2+\delta c_1\varepsilon+c_2\varepsilon}\|\nabla F\left(w^*\right)^{-1}\|<1,$$

and it follows from the Banach's Lemma 13 that B_k is nonsingular and

$$\|B_k^{-1}\| \le \Phi := \frac{\|\nabla F(w^*)^{-1}\|}{1-\sqrt{\delta^2+\delta c_1\varepsilon+c_2\varepsilon}\|\nabla F(w^*)^{-1}\|} \tag{7.19}$$

for some constant Φ. This proves assertion (ii).

(iii) Since B_k is nonsingular, Δw_k is well-defined and by (7.19)we have

$$\|\Delta w_k\| = \|B_k^{-1}(-F(w_k)+\mu_k\hat{e})\| \le \|B_k^{-1}\|(\|F(w_k)\|+\mu_k\|\hat{e}\|) = O(F(w_k))+O(\mu_k). \tag{7.20}$$

To prove (7.4), we note that if ε and δ are sufficiently small, then from the parameter condition $\mu_k = O\left(\|F\left(w_k\right)\|\right)$ and inequality (7.20), the assumptions (5.42) of Lemmas 20 are satisfied. By (5.43), we have

$$0 \le 1-\alpha_k \le (1-\gamma_k)+\Omega\|\Delta w_k\| = (1-\gamma_k)+O(F(w_k))+O(\mu_k).$$

This proves assertion (iii). □

7.4.2 Proof of Theorem 21

Proof. By mathematical induction, we will prove that if for $i=0,1,2,\ldots,k$,

$$d(w_i,w^*)<\varepsilon, \text{ and } \left\|G_i-\mathrm{T}_{\zeta_i^x}\operatorname{Hess}_x\mathcal{L}(w^*)\mathrm{T}_{\zeta_i^x}^{-1}\right\| \le \delta. \tag{7.21}$$

Then,

$$d(w_{k+1},w^*) \le \nu d(w_k,w^*)<\varepsilon, \text{ and } \left\|G_{k+1}-\mathrm{T}_{\zeta_{k+1}^x}\operatorname{Hess}_x\mathcal{L}(w^*)\mathrm{T}_{\zeta_{k+1}^x}^{-1}\right\| \le \delta, \tag{7.22}$$

Condition (7.21) clearly holds when $k=0$. If ε and δ are sufficiently small, it follows from (ii) of Lemma 25 that B_k is nonsingular and $\|B_k^{-1}\| \le \Phi$ with a positive constant. From the linear system (7.2), we have

$$\Delta w_k = B_k^{-1}\left(-F(w_k)+\mu_k\hat{e}\right). \tag{7.23}$$

Noting that $w^* = \bar{\mathrm{R}}_{w_k}\left(\xi_k\right)$ where $\xi_k := \bar{\mathrm{R}}_{w_k}^{-1}\left(w^*\right)$, $w_{k+1} = \bar{\mathrm{R}}_{w_k}\left(\alpha_k\Delta w_k\right)$, and (ii) of Lemma 6, we obtain

$$d(w_{k+1},w^*) \le a_1\left\|\alpha_k\Delta w_k-\xi_k\right\| = a_1\|\alpha_k B_k^{-1}\left(-F(w_k)+\mu_k\hat{e}\right)-\xi_k\|. \tag{7.24}$$

Let $r := \alpha_k B_k^{-1}\left(-F(w_k) + \mu_k \hat{e}\right) - \xi_k$. By $\bar{\mathrm{T}}_{\xi_k}^{-1} F(w^*) = 0$ and $\zeta_k := \bar{\mathrm{R}}_{w_k}^*(w^{-1})$, we have

$$\begin{aligned} r = &(\alpha_k - 1)\,\xi_k + \alpha_k \mu_k B_k^{-1}\hat{e} \alpha_k B_k^{-1}\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\right)\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1} F(w_k) \\ &+ \alpha_k \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1}\left(\bar{\mathrm{T}}_{\xi_k}^{-1} F\left(w^*\right) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\xi_k\right). \end{aligned} \tag{7.25}$$

Let θ_1 and θ_2 respectively denote the last two terms of the above equality, i.e.,

$$\begin{aligned} \theta_1 &:= \alpha_k B_k^{-1}\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\right)\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1} F(w_k), \\ \theta_2 &:= \alpha_k \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1}\left(\bar{\mathrm{T}}_{\xi_k}^{-1} F\left(w^*\right) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\xi_k\right). \end{aligned}$$

Then, by (i) of Lemma 25,

$$\begin{aligned} \|\theta_1\| &\le \alpha_k \|B_k^{-1}\| \left\|B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\right\| \left\|\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1}\right\| \|F(w_k)\| \\ &\le \Phi\sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon}\,\|\nabla F(w^*)^{-1}\|\|F(w_k)\| \text{ (since } \overline{\mathrm{T}} \text{ is isometric)} \\ &= \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} O(d(w_k, w^*)) \text{ (by Lemma 7)}, \end{aligned} \tag{7.26}$$

and by (ii) of Lemma 12,

$$\begin{aligned} \|\theta_2\| &\le \alpha_k \left\|\bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)^{-1}\bar{\mathrm{T}}_{\zeta_k}^{-1}\right\| \left\|\bar{\mathrm{T}}_{\xi_k}^{-1} F\left(w^*\right) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}_{\zeta_k}^{-1}\xi_k\right\| \\ &\le c_3 \|\nabla F\left(w^*\right)^{-1}\| \,\|\xi_k\|\, d(w_k, w^*) \\ &= O(d^2(w_k, w^*)) \text{ (by (iv) of Lemma 6)}. \end{aligned} \tag{7.27}$$

Since $d(w_{k+1}, w^*) \le a_1 \|r\|$ by (7.24), it follows from (7.25) to (7.27) and (iii) of Lemma 25 that

$$\begin{aligned} d(w_{k+1}, w^*) &\le a_1(1 - \alpha_k)\|\xi_k\| + O(\mu_k) + O(d^2(w_k, w^*)) + \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} O(d(w_k, w^*)) \\ &\le \left[(1 - \gamma_k) + O(F(w_k)) + O(\mu_k)\right] \frac{a_1}{a_0} d(w_k, w^*) \\ &\quad + O(\mu_k) + O(d^2(w_k, w^*)) + \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} O(d(w_k, w^*)) \\ &\le \frac{a_1}{a_0}(1 - \gamma_k) d(w_k, w^*) + [O(F(w_k)) + O(\mu_k)] O(d(w_k, w^*)) \\ &\quad + O(\mu_k) + O(d^2\left(w_k, w^*\right)) + \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} O(d(w_k, w^*)) \\ &\le \frac{a_1}{a_0}(1 - \hat{\gamma}) d(w_k, w^*) + \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} O(d(w_k, w^*)) + \varepsilon^{\min(1,\tau)} O(d(w_k, w^*)) \\ &= \left\{\frac{1 + \bar{\varepsilon}}{1 - \bar{\varepsilon}}(1 - \hat{\gamma}) + M_1 \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} + M_2 \varepsilon^{\min(1,\tau)}\right\} d(w_k, w^*), \end{aligned}$$

where $a_0 = 1 - \bar{\varepsilon}, a_1 = 1 + \bar{\varepsilon}$ for any $\bar{\varepsilon} > 0$ by (i) of Lemma 6, M_1 and M_2 are constants. Let

$$\psi(\bar{\varepsilon}, \varepsilon, \delta) := \left\{\frac{1 + \bar{\varepsilon}}{1 - \bar{\varepsilon}}(1 - \hat{\gamma}) + M_1 \sqrt{\delta^2 + \delta c_1 \varepsilon + c_2 \varepsilon} + M_2 \varepsilon^{\min(1,\tau)}\right\}.$$

Then,

$$\lim_{\bar{\varepsilon} \to 0^+, \varepsilon \to 0^+, \delta \to 0^+} \psi(\bar{\varepsilon}, \varepsilon, \delta) = 1 - \hat{\gamma} < \nu.$$

By choosing $\bar{\varepsilon}, \varepsilon$, and δ such that $\psi(\bar{\varepsilon}, \varepsilon, \delta) < \nu$, we obtain $d(w_{k+1}, w^*) \leq \nu d(w_k, w^*) < \varepsilon$.

The second part of the induction uses the same technique as in [39, Theorem 3.2]. For $i = 0, 1, 2, \ldots, k$,

$$\begin{aligned}
&\left\|G_{i+1} - \mathrm{T}_{\zeta^x_{i+1}} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_{i+1}}\right\| - \left\|G_i - \mathrm{T}_{\zeta^x_i} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_i}\right\| \\
&\leq \beta_1 \left\|G_i - \mathrm{T}_{\zeta^x_i} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_i}\right\| \sigma_i + \beta_2 \sigma_i \text{ (by (7.5))} \\
&\leq (\beta_1 \delta + \beta_2)\, \sigma_i \text{ (by (7.21))} \\
&\leq (\beta_1 \delta + \beta_2)\, \nu^i \varepsilon.
\end{aligned}$$

By summing both sides from $i = 0$ to $i = k$, we obtain

$$\left\|G_{k+1} - \mathrm{T}_{\zeta^x_{k+1}} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_{k+1}}\right\| \leq \left\|G_0 - \mathrm{T}_{\zeta^x_0} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_0}\right\| + \frac{(\beta_1 \delta + \beta_2)\, \varepsilon}{1 - \nu}.$$

By choosing ε and δ such that $\frac{(\beta_1\delta+\beta_2)\varepsilon}{1-\nu} < \frac{1}{2}\delta$, we obtain

$$\left\|G_{k+1} - \mathrm{T}_{\zeta^x_{k+1}} \operatorname{Hess}_x \mathcal{L}(w^*)\mathrm{T}^{-1}_{\zeta^x_{k+1}}\right\| \leq \delta.$$

The proof is complete. □

7.4.3 Proof of Theorem 22

Proof. Let us show an auxiliary result:

$$\|F(w_k)\| = O(\|\Delta w_k\|).$$

This comes from that the sequence $\{w_k\}$ converges linearly to the solution w^*. Note that linear convergence implies, for some $\nu \in (0, 1)$,

$$d(w_k, w^*) \leq d(w_k, w_{k+1}) + d(w_{k+1}, w^*) \leq d(w_k, w_{k+1}) + \nu d(w_k, w^*).$$

Thus, we have

$$\frac{\mathrm{d}(w_k, w^*)}{d(w_k, w_{k+1})} \leq \frac{1}{1 - \nu}.$$

Since $w_{k+1} = \bar{\mathrm{R}}_{w_k}(\alpha_k \Delta w_k)$, by (iii) of Lemma 6 we have $d(w_k, w_{k+1}) \leq a_1 \|\alpha_k \Delta w_k\|$. By Lemma 7, $\|F(w_k)\| \leq a_3 d(w_k, w^*)$. Finally, we see that

$$\frac{\|F(w_k)\|}{\|\Delta w_k\|} = \frac{\alpha_k a_1 \|F(w_k)\|}{a_1 \|\alpha_k \Delta w_k\|} \leq \frac{\alpha_k a_1 a_3 d(w_k, w^*)}{d(w_k, w_{k+1})} \leq \frac{a_1 a_3}{1 - \nu}, \tag{7.28}$$

and that $\|F(w_k)\| = O(\|\Delta w_k\|)$.

Next, we show that

$$\alpha_k \to 1. \tag{7.29}$$

Since we have assumed that $\{\|B_k^{-1}\|\}$ is bounded above, we have

$$\begin{aligned}\|\Delta w_k\| &= \left\|B_k^{-1}(-F(w_k)+\mu_k\hat{e})\right\| \text{ (by (7.2))}\\ &\le \|B_k^{-1}\|(\|F(w_k)\|+\mu_k\|\hat{e}\|) = O(F(w_k))+O(\mu_k) = O(F(w_k)) \text{ (by (7.7) for } \mu_k),\end{aligned} \tag{7.30}$$

and (7.30) together with $w_k \to w^*$ implies that the conditions of Lemmas 20 are satisfied,

$$0 \le 1-\alpha_k \le (1-\gamma_k)+\Omega\,\|\Delta w_k\| = (1-\gamma_k)+O(F(w_k)).$$

Thus, $\gamma_k \to 1$ implies $\alpha_k \to 1$.

(a $\Leftrightarrow$ b). Let (a) hold. By $B_k\Delta w_k = \mu_k\hat{e} - F(w_k)$, we have

$$\begin{aligned}\bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1}) = \bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1}) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\alpha_k\Delta w_k\\ -\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\right)\alpha_k\Delta w_k + (1-\alpha_k)\,F(w_k) + \alpha_k\mu_k\hat{e},\end{aligned} \tag{7.31}$$

where $\zeta_k = \mathrm{R}^{-1}_{w^*}(w_k)$. Thus, by Lemma 12, we see that

$$\begin{aligned}&\left\|\bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1})\right\|\\ &\le \left\|\bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1}) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\alpha_k\Delta w_k\right\|\\ &\quad + \left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\right)\alpha_k\Delta w_k\right\| + (1-\alpha_k)\,\|F(w_k)\| + \alpha_k\mu_k\,\|\hat{e}\|\\ &\le c_3\,\|\alpha_k\Delta w_k\|\max\{d(w_k,w^*),d(w_{k+1},w^*)\} + \left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\right)\alpha_k\Delta w_k\right\|\\ &\quad + (1-\alpha_k)\,\|F(w_k)\| + \alpha_k\mu_k\,\|\hat{e}\|,\end{aligned}$$

and by dividing both sides by $\|\alpha_k\Delta w_k\|$, we get

$$\begin{aligned}\frac{\left\|\bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1})\right\|}{\|\alpha_k\Delta w_k\|} &\le c_3\max\{d(w_k,w^*),d(w_{k+1},w^*)\} + \frac{\left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\right)\Delta w_k\right\|}{\|\Delta w_k\|}\\ &\quad + \left(\frac{1}{a_k}-1\right)\frac{\|F(w_k)\|}{\|\Delta w_k\|} + \frac{\mu_k}{\|F(w_k)\|}\frac{\|F(w_k)\|}{\|\Delta w_k\|}\,\|\hat{e}\|.\end{aligned}$$

Note that $\frac{\|F(w_k)\|}{\|\Delta w_k\|}$ is bounded by (7.28) and $\mu_k = o(\|F(w_k)\|)$. Taking the limit of the above gives (b).

Conversely, let (b) hold. From (7.31), we have

$$\begin{aligned}\left(B_k - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\right)\alpha_k\Delta w_k = \bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1}) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k}\nabla F\left(w^*\right)\bar{\mathrm{T}}^{-1}_{\zeta_k}\alpha_k\Delta w_k\\ - \bar{\mathrm{T}}^{-1}_{\alpha_k\Delta w_k}F(w_{k+1}) + (1-\alpha_k)\,F(w_k) + \alpha_k\mu_k\hat{e}.\end{aligned}$$

Again, taking the norm and using Lemma 12, we get

$$\begin{aligned}\left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F\left(w^*\right) \bar{\mathrm{T}}_{\zeta_k}^{-1}\right) \alpha_k \Delta w_k\right\| &\leq \left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) - F(w_k) - \bar{\mathrm{T}}_{\zeta_k} \nabla F\left(w^*\right) \bar{\mathrm{T}}_{\zeta_k}^{-1} \alpha_k \Delta w_k\right\| \\ &\quad + \left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| + (1-\alpha_k)\left\|F(w_k)\right\| + \alpha_k \mu_k \|\hat{e}\| \\ &\leq c_3 \left\|\alpha_k \Delta w_k\right\| \max\{d(w_k, w^*), d(w_{k+1}, w^*)\} \\ &\quad + \left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| + (1-\alpha_k)\left\|F(w_k)\right\| + \alpha_k \mu_k \|\hat{e}\|.\end{aligned}$$

Dividing both sides by $\|\alpha_k \Delta w_k\|$ and taking the limit gives (a) for the same reason described above.

(b ⇔ c). This part mainly uses Lemma 8. Let (b) hold. Observe that

$$\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) = \bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) - \mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) + \mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}). \tag{7.32}$$

Thus, Lemma 8 and the isometry of parallel transport $\mathrm{P}_{\alpha_k \Delta w_k}^{-1}$ show that

$$\begin{aligned}\left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| &\geq \left\|\mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| - \left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) - \mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| \\ &\geq \|F(w_{k+1})\| - a_4 \|F(w_{k+1})\| \|\alpha_k \Delta w_k\|,\end{aligned}$$

for some constant a_4, and hence,

$$\frac{\left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\|}{\|\alpha_k \Delta w_k\|} + a_4 \|F(w_{k+1})\| \geq \frac{\|F(w_{k+1})\|}{\|\alpha_k \Delta w_k\|} \geq \frac{\|F(w_{k+1})\|}{\|\Delta w_k\|}.$$

Since $\|F(w_k)\| \to 0$, taking the limit of the above gives (c).

Conversely, let (c) hold. Again, (7.32), Lemma 8, and the isometry of parallel transport yield

$$\begin{aligned}\left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| &\leq \left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1}) - \mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| + \left\|\mathrm{P}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\| \\ &\leq a_4 \|F(w_{k+1})\| \|\alpha_k \Delta w_k\| + \|F(w_{k+1})\|,\end{aligned}$$

and hence,

$$\frac{\left\|\bar{\mathrm{T}}_{\alpha_k \Delta w_k}^{-1} F(w_{k+1})\right\|}{\|\alpha_k \Delta w_k\|} \leq a_4 \|F(w_{k+1})\| + \frac{\|F(w_{k+1})\|}{\|\Delta w_k\|} \frac{1}{\alpha_k}.$$

Since $\|F(w_k)\| \to 0$ and $\alpha_k \to 1$, taking the limit of the above gives (b).

(c ⇔ d). Note that the statement that $\{w_k\}$ converges superlinearly to w^* can be rewritten equivalently as

$$\lim_{k \to \infty} \frac{\|\zeta_{k+1}\|}{\|\zeta_k\|} = 0,$$

where $\zeta_k \in T_{w^*}\mathcal{N}$ is defined by $\zeta_k = \mathrm{R}_{w^*}^{-1}(w_k)$. We will show that the above equation holds if and only if (c) holds.

This part mainly uses Corollary 1 and (iv) of Lemma 6. Let (c) hold. By (iv) of Lemma 6, it follows that

$$a_0 \|\alpha_k \Delta w_k\| \leq d(w_k, w_{k+1}) \leq d(w_k, w^*) + d(w_{k+1}, w^*) \leq a_1 \left(\|\zeta_k\| + \|\zeta_{k+1}\|\right),$$

thus,

$$\|\alpha_k \Delta w_k\| \leq \frac{a_1}{a_0} \left(\|\zeta_k\| + \|\zeta_{k+1}\|\right). \tag{7.33}$$

On the other hand, by Corollary 1,

$$a_4 \|\zeta_{k+1}\| \leq \|F(w_{k+1})\| \leq a_5 \|\zeta_{k+1}\|. \tag{7.34}$$

Therefore, by (7.33) and (7.34), we obtain

$$\frac{\|F(w_{k+1})\|}{\|\Delta w_k\|} = \alpha_k \frac{\|F(w_{k+1})\|}{\|\alpha_k \Delta w_k\|} \geq \alpha_k \frac{a_0 a_4 \|\zeta_{k+1}\|}{a_1 \left(\|\zeta_k\| + \|\zeta_{k+1}\|\right)} = \alpha_k \frac{a_0 a_4}{a_1} \cdot \frac{\|\zeta_{k+1}\| / \|\zeta_k\|}{\left(1 + \|\zeta_{k+1}\| / \|\zeta_k\|\right)}.$$

By (7.29), taking the limit of the above gives (d).

Conversely, let (d) hold. Again by (iv) of Lemma 6, similarly to the proof of [81, Theorem 14.1, Page 292], we find that

$$\|\alpha_k \Delta w_k\| \geq \frac{1}{a_1} d(w_k, w_{k+1}) \geq \frac{1}{a_1} \left(d(w_k, w^*) - d(w_{k+1}, w^*)\right) \geq \frac{a_0}{a_1} \left(\|\zeta_k\| - \|\zeta_{k+1}\|\right).$$

It follows that

$$\frac{\|F(w_{k+1})\|}{\|\Delta w_k\|} = \alpha_k \frac{\|F(w_{k+1})\|}{\|\alpha_k \Delta w_k\|} \leq \alpha_k \frac{a_1^2 \|\zeta_{k+1}\|}{a_0 \left(\|\zeta_k\| - \|\zeta_{k+1}\|\right)} = \alpha_k \frac{a_1^2}{a_0} \cdot \frac{\|\zeta_{k+1}\| / \|\zeta_k\|}{\left(1 - \|\zeta_{k+1}\| / \|\zeta_k\|\right)}.$$

Taking the limit of the above gives (c). We complete the proof. □

7.4.4 Proof of Theorem 23

Proof. Recall that $\Delta w_k = (\Delta x_k, \Delta y_k, \Delta s_k, \Delta z_k) \in T_{w_k}\mathcal{N}$. By Theorem 21 and (7.5), the sequence $\{w_k\}$ locally converges to w^* and $\|G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1}\|$ is bounded for all $k \geq 0$. From inequality (7.16) in the proof of Lemma 25, we conclude that

$$\begin{aligned}
&\left\| \left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F(w^*) \bar{\mathrm{T}}_{\zeta_k}^{-1} \right) \Delta w_k \right\|^2 \\
\leq &\left\| \left(G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right) \Delta x_k \right\|^2 \\
&+ \left\| G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right\| \|\Delta x_k\| O\left(d(w_k, w^*)\right) \|\Delta w_k\| \\
&+ O\left(d(w_k, w^*)\right) \|\Delta w_k\|^2 \\
\leq &\left\| \left(G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right) \Delta x_k \right\|^2 + O\left(d(w_k, w^*)\right) \|\Delta w_k\|^2.
\end{aligned} \tag{7.35}$$

The last inequality comes from

$$\begin{aligned}
&\left\| G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1} \right\| \|\Delta x_k\| O\left(d(w_k, w^*)\right) \|\Delta w_k\| \\
= &\|\Delta x_k\| O\left(d(w_k, w^*)\right) \|\Delta w_k\| \text{ (by boundedness of } \|G_k - \mathrm{T}_{\zeta_k^x} \mathrm{Hess}_x \mathcal{L}(w^*) \mathrm{T}_{\zeta_k^x}^{-1}\|) \\
\leq &O\left(d(w_k, w^*)\right) \|\Delta w_k\|^2 \text{ (by } \|\Delta x_k\| \leq \|\Delta w_k\|).
\end{aligned}$$

Thus, by dividing both sides of (7.35) by $\|\Delta w_k\|^2$, we have

$$\begin{aligned}\frac{\left\|\left(B_k - \bar{\mathrm{T}}_{\zeta_k} \nabla F\left(w^*\right) \bar{\mathrm{T}}_{\zeta_k}^{-1}\right) \Delta w_k\right\|^2}{\|\Delta w_k\|^2} &= \frac{\left\|\left(G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}\left(w^*\right) \mathrm{T}_{\zeta_k^x}^{-1}\right) \Delta x_k\right\|^2}{\|\Delta w_k\|^2} + O(d(w_k, w^*)) \\ &\leq \frac{\left\|\left(G_k - \mathrm{T}_{\zeta_k^x} \operatorname{Hess}_x \mathcal{L}\left(w^*\right) \mathrm{T}_{\zeta_k^x}^{-1}\right) \Delta x_k\right\|^2}{\|\Delta x_k\|^2} + O(d(w_k, w^*)).\end{aligned}$$

Taking the limit above with (7.9) and invoking Theorem 22 complete the proof. □

Chapter 8

Conclusions and Future Research

8.1 Conclusions

Riemannian optimization is an important research area in the field of optimization theory. Existing research on Riemannian algorithms has focused on the problems with smooth objective functions on a single manifold. In this thesis, two variants of Riemannian optimization are investigate, each of which solves a different challenge in practical applications.

The first variant is Nonsmooth Riemannian Optimization (NRO), which is concerned with optimization problems with nonsmooth objective functions. For NRO problems, we propose a generalized framework of Riemannian smoothing method, ensuring efficient convergence to the limiting stationary point. Our framework facilitates the use of existing solvers such as Manopt, thus enabling fast code implementation. In particular, we apply our method to the completely positive matrix factorization problem. Numerical experiments confirm that our method is particularly suitable for large-scale factorization problems.

The second variant is Constrained Riemannian Optimization (CRO), which deals with optimization problems on non-single manifolds, i.e., problems involving additional constraints. For CRO problems, we propose a Riemannian version of the classical interior point method, namely the Riemannian Interior Point Method (RIPM), and establish its local and global convergence. To our knowledge, this is the first study to apply the primal-dual interior point method on a Riemannian manifold. Numerical experiments show the stability and efficiency of our method. To conclude this subsection, let us compare with the existing interior point method, i.e., the so-called Euclidean Interior Point Method (EIPM), to illustrate the theoretical advantages of our RIPM.

Comparison: Riemannian Interior Point Methods (RIPM) v.s. Euclidean Interior Point Methods (EIPM).

1. RIPM generalizes EIPM form Euclidean space to general Riemannian manifolds. EIPM is a special case of RIPM when $\mathcal{M} = \mathbb{R}^n$ or $\mathbb{R}^{m\times n}$ in (CRO).
2. RIPM inherits the all advantages of Riemannian optimization. For example, we can exploit the geometric structure of $\mathcal{M}$, which is usually regarded as a set of constraints from the Euclidean viewpoint.
3. Note that in both RIPM and EIPM, we have to solve the condensed Newton equation (5.32) at each iteration. However, if the equality constraints can be considered to be a manifold, RIPM can solve

(5.32) with a *smaller* order on $T_x\mathcal{M} \times \mathbb{R}^l$. For example, the problem (Model_St) can be rewritten as

$$\min_{X \in \mathbb{R}^{n\times k}} -2\,\mathrm{trace}(X^T C) \quad \text{s.t. } X^T X = I_k,\ X \geq 0.$$

Here, Stiefel manifold is replaced by the equality constraints, i.e., we define $h\colon \mathbb{R}^{n\times k} \to \mathrm{Sym}(k) : X \mapsto h(X) := X^T X - I_k$; and $\mathcal{M} = \mathbb{R}^{n\times k}$, $l = \dim \mathrm{Sym}(k) = k(k+1)/2$ in (CRO). Then, when we apply EIPM, it requires us to solve (5.32) of order $nk + k(k+1)/2$. On the other hand, if we apply RIPM to (Model_St), then (5.32) reduces to (5.35) since there are only inequality constraints on $\mathcal{M} = \mathrm{St}(n,k)$. In this case, we solve the equation of order $nk - k(k+1)/2$, i.e., the dimension of $\mathrm{St}(n,k)$. Compared to EIPM, using RIPM reduces our dimensionality by $k(k+1)$.

4. RIPM can solve some problems that EIPM cannot. For example, the problem (NLRM) can be rewritten as

$$\min_{X \in \mathbb{R}^{m\times n}} \|A - X\|_{\mathrm{F}}^2 \quad \text{s.t. } \mathrm{rank}(X) = r,\ X \geq 0.$$

Since the rank function, $X \mapsto \mathrm{rank}(X)$, is not even continuous, we cannot apply EIPM.

8.2 Future Research

Finally, we discuss several promising research topics. Since the work in this thesis on the extension of the interior point method on manifolds is the first of its kind, much meaningful work remains to be explored. Considering the maturity of the existing Euclidean interior point methods in all aspects of numerical implementation and algorithmic theory, it is well worthwhile to continue exploring these methods on manifolds in the future.

Preconditioner for linear operator equation. Due to the complementary condition, as $k \to \infty$, the values of $S_k^{-1} Z_k$ display a huge difference in magnitude: some of them tend to zero while others go to infinity. Hence, the presence of the operator $\Psi := G_x S^{-1} Z G_x^*$ in the system (5.32) makes it very ill-conditioned, so the iterative method will likely fail unless it is carefully preconditioned. Unfortunately, operator equation has no explicit matrix form available, which makes the most common preconditioner methods based on matrix decomposition techniques useless. A possible way around this is to find another nonsingular operator $\mathcal{P}$ such that the condition number of the new operator $\mathcal{P}^{-1}\mathcal{T}$ becomes smaller.

Treatment of more state-of-the-art interior point methods. While we have considered interior point methods on a manifold for the first time, our Euclidean theoretic counterpart is an early nonlinear interior point method algorithm [73]; however, the counterpart now appears to be obsolete compared with more recent interior point methods. For example, our method does not drive the iteration towards minimizers, but only towards stationary points; globalization is done by monitoring only the KKT residuals; moreover, the boundedness assumption (C2) of $\{z_k\}$ is too strong to hold in some simple cases (see Wächter-Biegler effect [199]). It remains an important issue to adapt more modern interior point methods to manifolds, although we may encounter various difficulties in Riemannian geometry.

Bibliography

[1] Absil, P.-A., Amodei, L., and Meyer, G. (2014). Two Newton methods on the manifold of fixed-rank matrices endowed with Riemannian quotient geometries. *Computational Statistics*, 29:569–590.

[2] Absil, P.-A., Baker, C. G., and Gallivan, K. A. (2007). Trust-region methods on Riemannian manifolds. *Foundations of Computational Mathematics*, 7:303–330.

[3] Absil, P.-A. and Gallivan, K. A. (2006). Joint diagonalization on the oblique manifold for independent component analysis. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 5, pages V–V. IEEE.

[4] Absil, P.-A. and Hosseini, S. (2019). A collection of nonsmooth Riemannian optimization problems. In Hosseini, S., Mordukhovich, B. S., and Uschmajew, A., editors, *Nonsmooth Optimization and Its Applications*, pages 1–15. Springer International Publishing, Cham. Series Title: International Series of Numerical Mathematics.

[5] Absil, P.-A., Mahony, R., and Sepulchre, R. (2008). *Optimization Algorithms on Matrix Manifolds*. Princeton University Press.

[6] Adler, R. L., Dedieu, J.-P., Margulies, J. Y., Martens, M., and Shub, M. (2002). Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA Journal of Numerical Analysis*, 22(3):359–390.

[7] Aihara, K. and Sato, H. (2017). A matrix-free implementation of Riemannian Newton's method on the Stiefel manifold. *Optimization Letters*, 11(8):1729–1741.

[8] Altschuler, J., Chewi, S., Gerber, P. R., and Stromme, A. (2021). Averaging on the Bures-Wasserstein manifold: Dimension-free convergence of gradient descent. In *Advances in Neural Information Processing Systems*, volume 34.

[9] Axler, S. (2015). *Linear Algebra Done Right*. Undergraduate Texts in Mathematics. Springer International Publishing, Cham.

[10] Bagirov, A., Karmitsa, N., and Mäkelä, M. M. (2014). *Introduction to Nonsmooth Optimization: Theory, Practice and Software*. Springer International Publishing, Cham.

[11] Baker, C. G. (2008). *Riemannian manifold trust-region methods with applications to eigenproblems*. PhD thesis, Florida State University.

[12] Bendokat, T., Zimmermann, R., and Absil, P.-A. (2020). A Grassmann manifold handbook: Basic geometry and computational aspects. arXiv:2011.13699.

[13] Bento, G. C., Ferreira, O. P., and Melo, J. G. (2017). Iteration-complexity of gradient, subgradient and proximal point methods on Riemannian manifolds. *Journal of Optimization Theory and Applications*, 173:548–562.

[14] Bento, G. D. C., Da Cruz Neto, J. X., and De Meireles, L. V. (2018). Proximal point method for locally Lipschitz functions in multiobjective optimization of Hadamard manifolds. *Journal of Optimization Theory and Applications*, 179:37–52.

[15] Benzi, M., Golub, G. H., and Liesen, J. (2005). Numerical solution of saddle point problems. *Acta numerica*, 14:1–137.

[16] Bergmann, R. (2022). Manopt.jl: Optimization on manifolds in Julia. *Journal of Open Source Software*, 7(70):3866.

[17] Bergmann, R. and Herzog, R. (2019). Intrinsic formulation of KKT conditions and constraint qualifications on smooth manifolds. *SIAM Journal on Optimization*, 29(4):2423–2444.

[18] Berman, A., Dür, M., and Shaked-Monderer, N. (2015). Open problems in the theory of completely positive and copositive matrices. *Electronic Journal of Linear Algebra*, 29:46–58.

[19] Berman, A. and Shaked-Monderer, N. (2003). *Completely Positive Matrices*. World Scientific.

[20] Bertsekas, D. P. (2016). *Nonlinear Programming*. Athena Scientific, Belmont, 3 edition.

[21] Bhatia, R. (2009). *Positive Definite Matrices*. Princeton university press.

[22] Bian, W. and Chen, X. (2014). Neural network for nonsmooth, nonconvex constrained minimization via smooth approximation. *IEEE Transactions on Neural Networks and Learning Systems*, 25(3):545–556.

[23] Bomze, I. M. (2012). Copositive optimization — Recent developments and applications. *European Journal of Operational Research*, 216(3):509–520.

[24] Bomze, I. M. (2018). Building a completely positive factorization. *Central European Journal of Operations Research*, 26:287–305.

[25] Bomze, I. M., Dickinson, P. J., and Still, G. (2015). The structure of completely positive matrices according to their CP-rank and CP-plus-rank. *Linear Algebra and Its Applications*, 482:191–206.

[26] Bomze, I. M., Dür, M., De Klerk, E., Roos, C., Quist, A. J., and Terlaky, T. (2000). On copositive programming and standard quadratic optimization problems. *Journal of Global Optimization*, 18:301–320.

[27] Bomze, I. M., Schachinger, W., and Uchida, G. (2012). Think co(mpletely)positive! Matrix properties, examples and a clustered bibliography on copositive optimization. *Journal of Global Optimization*, 52:423–445.

[28] Bonettini, S., Galligani, E., and Ruggiero, V. (2005). An inexact Newton method combined with Hestenes multipliers' scheme for the solution of Karush-Kuhn-Tucker systems. *Applied Mathematics and Computation*, 168:651–676.

[29] Bonnabel, S. (2013). Stochastic gradient descent on Riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9):2217–2229.

[30] Borckmans, P. B., Ishteva, M., and Absil, P.-A. (2010). A modified particle swarm optimization algorithm for the best low multilinear rank approximation of higher-order tensors. In Dorigo, M., Birattari, M., Di Caro, G. A., Doursat, R., Engelbrecht, A. P., Floreano, D., Gambardella, L. M., Groß, R., Şahin, E., Sayama, H., and Stützle, T., editors, *Swarm Intelligence*, pages 13–23, Berlin, Heidelberg. Springer Berlin Heidelberg.

[31] Borckmans, P. B., Selvan, S. E., Boumal, N., and Absil, P.-A. (2014). A Riemannian subgradient algorithm for economic dispatch with valve-point effect. *Journal of Computational and Applied Mathematics*, 255:848–866.

[32] Bortoloti, M. A. D. A., Fernandes, T. A., Ferreira, O. P., and Yuan, J. (2020). Damped Newton's method on Riemannian manifolds. *Journal of Global Optimization*, 77:643–660.

[33] Boţ, R. I. and Nguyen, D.-K. (2021). Factorization of completely positive matrices using iterative projected gradient steps. *Numerical Linear Algebra with Applications*, 28(6):e2391.

[34] Boumal, N. (2023). *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 1 edition.

[35] Boumal, N. and Absil, P.-a. (2011). RTRMC: A Riemannian trust-region method for low-rank matrix completion. In *Advances in Neural Information Processing Systems*, volume 24, pages 406–414.

[36] Boumal, N., Absil, P.-A., and Cartis, C. (2019). Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33.

[37] Boumal, N., Mishra, B., Absil, P.-A., and Sepulchre, R. (2014). Manopt, a Matlab toolbox for optimization on manifolds. *Journal of Machine Learning Research*, 15(42):1455–1459.

[38] Brockett, R. W. (1993). Differential geometry and the design of gradient algorithms. *Proc. Symp. Pure Math*, 54(0):69–92.

[39] Broyden, C. G., Dennis Jr, J. E., and Moré, J. J. (1973). On the local and superlinear convergence of quasi-Newton methods. *IMA Journal of Applied Mathematics*, 12(3):223–245.

[40] Burer, S. (2009). On the copositive representation of binary and continuous nonconvex quadratic programs. *Mathematical Programming*, 120:479–495.

[41] Burer, S. (2015). A gentle, geometric introduction to copositive optimization. *Mathematical Programming*, 151:89–116.

[42] Burke, J. V., Curtis, F. E., Lewis, A. S., Overton, M. L., and Simões, L. E. A. (2020). Gradient sampling methods for nonsmooth optimization. In Bagirov, A. M., Gaudioso, M., Karmitsa, N., Mäkelä, M. M., and Taheri, S., editors, *Numerical Nonsmooth Optimization: State of the Art Algorithms*, pages 201–225. Springer International Publishing, Cham.

[43] Cambier, L. and Absil, P.-A. (2016). Robust low-rank matrix completion by Riemannian optimization. *SIAM Journal on Scientific Computing*, 38(5):S440–S460.

[44] Cannon, J. W., Floyd, W. J., Kenyon, R., and Parry, W. R. (1997). Hyperbolic geometry. In *Flavors of Geometry*. Cambridge Univ. Press.

[45] Carson, T., Mixon, D. G., Villar, S., and Ward, R. (2017). Manifold optimization for k-means clustering. In *2017 International Conference on Sampling Theory and Applications (SampTA)*, pages 73–77. IEEE.

[46] Chami, I., Ying, Z., Ré, C., and Leskovec, J. (2019). Hyperbolic graph convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 32.

[47] Chattopadhyay, A., Selvan, S. E., and Amato, U. (2015). A derivative-free Riemannian Powell's method, minimizing Hartley-entropy-based ICA contrast. *IEEE Transactions on Neural Networks and Learning Systems*, 27(9):1983–1990.

[48] Chen, C., Pong, T. K., Tan, L., and Zeng, L. (2020a). A difference-of-convex approach for split feasibility with applications to matrix factorizations and outlier detection. *Journal of Global Optimization*, 78:107–136.

[49] Chen, F., Yang, Y., Xu, L., Zhang, T., and Zhang, Y. (2019). Big-data clustering: k-means or k-indicators? arXiv:1906.00938.

[50] Chen, M. (2023). Gram-Schmidt orthogonalization. MATLAB Central File Exchange. Retrieved September 9, 2023.

[51] Chen, S., Ma, S., Man-Cho So, A., and Zhang, T. (2020b). Proximal gradient method for nonsmooth optimization over the Stiefel manifold. *SIAM Journal on Optimization*, 30(1):210–239.

[52] Chen, X. (2012). Smoothing methods for nonsmooth, nonconvex minimization. *Mathematical Programming*, 134:71–99.

[53] Chen, X., Wets, R. J.-B., and Zhang, Y. (2012). Stochastic variational inequalities: Residual minimization smoothing sample average approximations. *SIAM Journal on Optimization*, 22(2):649–673.

[54] Cherian, A. and Sra, S. (2017). Riemannian dictionary learning and sparse coding for positive definite matrices. *IEEE Transactions on Neural Networks and Learning Systems*, 28(12):2859–2871.

[55] Chewi, S., Maunu, T., Rigollet, P., and Stromme, A. J. (2020). Gradient descent algorithms for Bures-Wasserstein barycenters. In Abernethy, J. and Agarwal, S., editors, *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pages 1276–1304. PMLR.

[56] De Carvalho Bento, G., Da Cruz Neto, J. X., and Oliveira, P. R. (2016). A new approach to the proximal point method: Convergence on general Riemannian manifolds. *Journal of Optimization Theory and Applications*, 168:743–755.

[57] De Klerk, E. and Pasechnik, D. V. (2002). Approximation of the stability number of a graph via copositive programming. *SIAM Journal on Optimization*, 12(4):875–892.

[58] Dennis, J. E. and Moré, J. J. (1974). A characterization of superlinear convergence and its application to quasi-Newton methods. *Mathematics of Computation*, 28(126):549–560.

[59] Dennis Jr, J. E. and Schnabel, R. B. (1996). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. SIAM.

[60] Dickinson, P. J. (2010). An improved characterisation of the interior of the completely positive cone. *Electronic Journal of Linear Algebra*, 20:723–729.

[61] Dickinson, P. J. and Dür, M. (2012). Linear-time complete positivity detection and decomposition of sparse matrices. *SIAM Journal on Matrix Analysis and Applications*, 33(3):701–720.

[62] Dickinson, P. J. and Gijben, L. (2014). On the computational complexity of membership problems for the completely positive cone and its dual. *Computational Optimization and Applications*, 57:403–415.

[63] Diepeveen, W., Lellmann, J., Öktem, O., and Schönlieb, C.-B. (2023). Regularizing orientation estimation in cryogenic electron microscopy three-dimensional map refinement through measure-based lifting over Riemannian manifolds. *SIAM Journal on Imaging Sciences*, 16(3):1440–1490.

[64] Do Carmo, M. P. and Flaherty Francis, J. (1992). *Riemannian Geometry*. Springer.

[65] Douik, A., Salehi, F., and Hassibi, B. (2019). A novel Riemannian optimization approach and algorithm for solving the phase retrieval problem. In *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pages 1962–1966. IEEE.

[66] Dreisigmeyer, D. W. (2007). Direct search algorithms over Riemannian manifolds. Optimization Online.

[67] Dür, M. (2010). Copositive programming — a survey. In Diehl, M., Glineur, F., Jarlebring, E., and Michiels, W., editors, *Recent Advances in Optimization and its Applications in Engineering*, pages 3–20, Berlin, Heidelberg. Springer Berlin Heidelberg.

[68] Dür, M. and Rendl, F. (2021). Conic optimization: A survey with special focus on copositive optimization and binary quadratic problems. *EURO Journal on Computational Optimization*, 9:100021.

[69] Dür, M. and Still, G. (2008). Interior points of the completely positive cone. *The Electronic Journal of Linear Algebra*, 17:48–53.

[70] Durazzi, C. (2000). On the Newton interior-point method for nonlinear programming problems. *Journal of Optimization Theory and Applications*, 104:73–90.

[71] Durazzi, C. and Ruggiero, V. (2004). Global convergence of the Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications*, 120:199–208.

[72] Edelman, A., Arias, T. A., and Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM journal on Matrix Analysis and Applications*, 20(2):303–353.

[73] El-Bakry, A., Tapia, R. A., Tsuchiya, T., and Zhang, Y. (1996). On the formulation and theory of the Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications*, 89:507–541.

[74] Fernandes, T. A., Ferreira, O. P., and Yuan, J. (2017). On the superlinear convergence of Newton's method on Riemannian manifolds. *Journal of Optimization Theory and Applications*, 173:828–843.

[75] Ferreira, O. P., Louzeiro, M. S., and Prudente, L. F. (2019). Iteration-complexity of the subgradient method on Riemannian manifolds with lower bounded curvature. *Optimization*, 68(4):713–729.

[76] Ferreira, O. P. and Silva, R. C. (2012). Local convergence of Newton's method under a majorant condition in Riemannian manifolds. *IMA Journal of Numerical Analysis*, 32(4):1696–1713.

[77] Ferreira, O. P. and Svaiter, B. F. (2002). Kantorovich's theorem on Newton's method in Riemannian manifolds. *Journal of Complexity*, 18(1):304–329.

[78] Fong, R. S. and Tino, P. (2022). *Population-Based Optimization on Riemannian Manifolds*. Springer.

[79] Gabay, D. (1982). Minimizing a differentiable function over a differential manifold. *Journal of Optimization Theory and Applications*, 37:177–219.

[80] Gallier, J. Q. and Quaintance, J. (2020). *Differential Geometry and Lie Groups*. Springer, 1 edition.

[81] Gallivan, K. A., Qi, C., and Absil, P.-A. (2012). A Riemannian Dennis-Moré condition. In Berry, M. W., Gallivan, K. A., Gallopoulos, E., Grama, A., Philippe, B., Saad, Y., and Saied, F., editors, *High-Performance Scientific Computing*, pages 281–293. Springer London, London.

[82] Ganea, O., Bécigneul, G., and Hofmann, T. (2018). Hyperbolic neural networks. In *Advances in Neural Information Processing Systems*, volume 31.

[83] Gao, B. and Absil, P.-A. (2022). A Riemannian rank-adaptive method for low-rank matrix completion. *Computational Optimization and Applications*, 81:67–90.

[84] Gao, B., Peng, R., and Yuan, Y.-x. (2023a). Optimization on product manifolds under a preconditioned metric. arXiv:2306.08873.

[85] Gao, B., Peng, R., and Yuan, Y.-x. (2023b). Riemannian preconditioned algorithms for tensor completion via tensor ring decomposition. arXiv:2302.14456.

[86] Gohary, R. H. and Davidson, T. N. (2009). Noncoherent MIMO communication: Grassmannian constellations and efficient detection. *IEEE Transactions on Information Theory*, 55(3):1176–1205.

[87] Goldfarb, D., Wen, Z., and Yin, W. (2009). A curvilinear search method for p-harmonic flows on spheres. *SIAM Journal on Imaging Sciences*, 2(1):84–109.

[88] Gondzio, J. (2012). Interior point methods 25 years later. *European Journal of Operational Research*, 218(3):587–601.

[89] Groetzner, P. and Dür, M. (2020). A factorization method for completely positive matrices. *Linear Algebra and Its Applications*, 591:1–24.

[90] Grohs, P. and Hosseini, S. (2016a). Nonsmooth trust region algorithms for locally Lipschitz functions on Riemannian manifolds. *IMA Journal of Numerical Analysis*, 36(3):1167–1192.

[91] Grohs, P. and Hosseini, S. (2016b). ε-subgradient algorithms for locally Lipschitz functions on Riemannian manifolds. *Advances in Computational Mathematics*, 42:333–360.

[92] Han, A., Mishra, B., Jawanpuria, P., and Gao, J. (2022). Riemannian block SPD coupling manifold and its application to optimal transport. *Machine Learning*.

[93] Han, A., Mishra, B., Jawanpuria, P. K., and Gao, J. (2021). On Riemannian optimization over positive definite matrices with the Bures-Wasserstein geometry. In *Advances in Neural Information Processing Systems*, volume 34.

[94] Helmke, U. and Moore, J. B. (1994). *Optimization and Dynamical Systems*. Springer.

[95] Hiroshi, H., Nieuwboer, H., and Walter, M. (2023). Interior-point methods on manifolds: Theory and applications. In *2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 2021–2030.

[96] Horn, R. A. and Johnson, C. R. (2012). *Matrix Analysis*. Cambridge University Press, 2 edition.

[97] Hosseini, R. and Sra, S. (2015). Matrix manifold optimization for Gaussian mixtures. In *Advances in Neural Information Processing Systems*, volume 28.

[98] Hosseini, R. and Sra, S. (2020). An alternative to EM for Gaussian mixture models: Batch and stochastic Riemannian optimization. *Mathematical Programming*, 181:187–223.

[99] Hosseini, S., Huang, W., and Yousefpour, R. (2018). Line search algorithms for locally Lipschitz functions on Riemannian manifolds. *SIAM Journal on Optimization*, 28(1):596–619.

[100] Hosseini, S. and Uschmajew, A. (2017). A Riemannian gradient sampling algorithm for nonsmooth optimization on manifolds. *SIAM Journal on Optimization*, 27(1):173–189.

[101] Hu, J., Liu, X., Wen, Z.-W., and Yuan, Y.-X. (2020). A brief introduction to manifold optimization. *Journal of the Operations Research Society of China*, 8(2):199–248.

[102] Huang, W. (2013). *Optimization algorithms on Riemannian manifolds with applications*. PhD thesis, Florida State University.

[103] Huang, W. (2021). Heuristics for optimization with nonnegativity constraints using ideas from Riemannian optimization. *IFAC-PapersOnLine*, 54(9):552–557.

[104] Huang, W., Absil, P.-A., and Gallivan, K. A. (2015a). A Riemannian symmetric rank-one trust-region method. *Mathematical Programming*, 150:179–216.

[105] Huang, W., Absil, P.-A., and Gallivan, K. A. (2018a). A Riemannian BFGS method without differentiated retraction for nonconvex optimization problems. *SIAM Journal on Optimization*, 28(1):470–495.

[106] Huang, W., Absil, P.-A., Gallivan, K. A., and Hand, P. (2018b). ROPTLIB: An object-oriented C++ library for optimization on Riemannian manifolds. *ACM Transactions on Mathematical Software*, 44(4):1–21.

[107] Huang, W., Gallivan, K. A., and Absil, P.-A. (2015b). A Broyden class of quasi-Newton methods for Riemannian optimization. *SIAM Journal on Optimization*, 25(3):1660–1685.

[108] Huang, W., Gallivan, K. A., Srivastava, A., and Absil, P.-A. (2016a). Riemannian optimization for registration of curves in elastic shape analysis. *Journal of Mathematical Imaging and Vision*, 54:320–343.

[109] Huang, W., Gallivan, K. A., and Zhang, X. (2016b). Solving Phaselift by low-rank Riemannian optimization methods. *Procedia Computer Science*, 80:1125–1134.

[110] Huang, W., Gallivan, K. A., and Zhang, X. (2017). Solving PhaseLift by low-rank Riemannian optimization methods for complex semidefinite constraints. *SIAM Journal on Scientific Computing*, 39(5):B840–B859.

[111] Huang, W. and Wei, K. (2022). Riemannian proximal gradient methods. *Mathematical Programming*, 194:371–413.

[112] Huang, Z. and Van Gool, L. (2017). A Riemannian network for SPD matrix learning. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, volume 31.

[113] Huang, Z., Wu, J., and Van Gool, L. (2018c). Building deep networks on Grassmann manifolds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[114] Jarre, F. and Schmallowsky, K. (2009). On the computation of C^* certificates. *Journal of Global Optimization*, 45:281.

[115] Jiang, B. and Dai, Y.-H. (2015). A framework of constraint preserving update schemes for optimization on Stiefel manifold. *Mathematical Programming*, 153:535–575.

[116] Jiang, B., Meng, X., Wen, Z., and Chen, X. (2023). An exact penalty approach for optimization with nonnegative orthogonality constraints. *Mathematical Programming*, 198:855–897.

[117] Kantorovich, L. V. and Akilov, G. P. (1982). *Functional Analysis*. Pergamon, 2 edition.

[118] Kasai, H. and Mishra, B. (2016). Low-rank tensor completion: A Riemannian manifold preconditioning approach. In Balcan, M. F. and Weinberger, K. Q., editors, *Proceedings of the 33rd International Conference on Machine Learning*, volume 48, pages 1012–1021. PMLR.

[119] Kelley, C. T. (1995). *Iterative Methods for Linear and Nonlinear Equations*. Number 16 in Frontiers in applied mathematics. SIAM, Philadelphia.

[120] Kochurov, M., Karimov, R., and Kozlukov, S. (2020). Geoopt: Riemannian optimization in pytorch. arXiv:2005.02819.

[121] Kojima, M., Mizuno, S., and Yoshise, A. (1989). A primal-dual interior point algorithm for linear programming. In Megiddo, N., editor, *Progress in Mathematical Programming*, pages 29–47. Springer New York, New York, NY.

[122] Kovnatsky, A., Glashoff, K., and Bronstein, M. M. (2016). MADMM: A generic algorithm for non-smooth optimization on manifolds. In Leibe, B., Matas, J., Sebe, N., and Welling, M., editors, *Computer Vision – ECCV 2016*, pages 680–696, Cham. Springer International Publishing.

[123] Kressner, D., Steinlechner, M., and Vandereycken, B. (2014). Low-rank tensor completion by Riemannian optimization. *BIT Numerical Mathematics*, 54:447–468.

[124] Lai, R. and Osher, S. (2014). A splitting method for orthogonality constrained problems. *Journal of Scientific Computing*, 58:431–449.

[125] Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791.

[126] Lee, J. M. (2018). *Introduction to Riemannian Manifolds*. Springer, 2 edition.

[127] Lee, J. M. and Lee, J. M. (2012). *Introduction to Smooth Manifolds*. Springer, 2 edition.

[128] Liu, C. and Boumal, N. (2020). Simple algorithms for optimization on Riemannian manifolds with constraints. *Applied Mathematics & Optimization*, 82:949–981.

[129] Luchnikov, I., Ryzhov, A., Filippov, S., and Ouerdane, H. (2021). QGOpt: Riemannian optimization for quantum technologies. *SciPost Physics*, 10:79.

[130] Luenberger, D. G. (1972). The gradient projection method along geodesics. *Management Science*, 18(11):620–631.

[131] Luenberger, D. G. (1973). *Introduction to Linear and Nonlinear Programming*, volume 28. Addison-wesley Reading, MA.

[132] Lustig, I., Marsten, R. E., and Shanno, D. F. (1991). Computational experience with a primal-dual interior point method for linear programming. *Linear Algebra and Its Applications*, 152:191–222.

[133] Martin, S., Raim, A. M., Huang, W., and Adragni, K. P. (2020). ManifoldOptim: An R interface to the ROPTLIB library for Riemannian manifold optimization. *Journal of Statistical Software*, 93(1):1–32.

[134] Martínez, H. J., Parada, Z., and Tapia, R. A. (1995). On the characterization of Q-superlinear convergence of quasi-Newton interior-point methods for nonlinear programming. Technical Report TR94-08, Computational and Applied Mathematics Department, Rice University.

[135] Meghwanshi, M., Jawanpuria, P., Kunchukuttan, A., Kasai, H., and Mishra, B. (2018). McTorch, a manifold optimization library for deep learning. arXiv:1810.01811.

[136] Megiddo, N. (1989). Pathways to the optimal set in linear programming. In Megiddo, N., editor, *Progress in Mathematical Programming: Interior-Point and Related Methods*, pages 131–158. Springer New York, New York, NY.

[137] Miolane, N., Guigui, N., Le Brigant, A., Mathe, J., Hou, B., Thanwerdas, Y., Heyder, S., Peltre, O., Koep, N., Zaatiti, H., et al. (2020). Geomstats: A Python package for Riemannian geometry in machine learning. *Journal of Machine Learning Research*, 21(223):1–9.

[138] Mishra, B., Meyer, G., Bonnabel, S., and Sepulchre, R. (2014). Fixed-rank matrix factorizations and Riemannian low-rank optimization. *Computational Statistics*, 29:591–621.

[139] Mishra, B., Satyadev, N., Kasai, H., and Jawanpuria, P. (2021). Manifold optimization for non-linear optimal transport problems. arXiv:2103.00902.

[140] Mohades, M. M. and Kahaei, M. H. (2021). An efficient Riemannian gradient based algorithm for max-cut problems. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69(3):1882–1886.

[141] Montanari, A. and Richard, E. (2016). Non-negative principal component analysis: Message passing algorithms and sharp asymptotics. *IEEE Transactions on Information Theory*, 62(3):1458–1484.

[142] Newton, I. (1687). *Philosophiae Naturalis Principia Mathematica*. William Dawson & Sons Ltd., London.

[143] Nguyen, X. S., Brun, L., Lézoray, O., and Bougleux, S. (2019). A neural network based on SPD manifold learning for skeleton-based hand gesture recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12036–12045.

[144] Nickel, M. and Kiela, D. (2017). Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems*, volume 30.

[145] Nickel, M. and Kiela, D. (2018). Learning continuous hierarchies in the Lorentz model of hyperbolic geometry. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80. PMLR.

[146] Nie, J. (2014). The $\mathcal{A}$-truncated k-moment problem. *Foundations of Computational Mathematics*, 14:1243–1276.

[147] Nocedal, J. and Wright, S. (2006). *Numerical Optimization*. Springer Science & Business Media, 2 edition.

[148] Obara, M., Okuno, T., and Takeda, A. (2020). Sequential quadratic optimization for nonlinear optimization problems on Riemannian manifolds. *SIAM Journal on Optimization*, 32(2):822–853. Publisher: SIAM.

[149] Ortega, J. M. and Rheinboldt, W. C. (1970). *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York.

[150] O'Searcoid, M. (2006). *Metric Spaces*. Springer Science & Business Media.

[151] Ozoliņš, V., Lai, R., Caflisch, R., and Osher, S. (2013). Compressed modes for variational problems in mathematics and physics. *Proceedings of the National Academy of Sciences*, 110(46):18368–18373.

[152] Peng, W., Varanka, T., Mostafa, A., Shi, H., and Zhao, G. (2022). Hyperbolic deep neural networks: A survey. *IEEE Transactions on pattern analysis and machine intelligence*, 44(12):10023–10044.

[153] Qi, C. (2011). *Numerical optimization methods on Riemannian manifolds*. PhD thesis, Florida State University.

[154] Qi, G., Yu, H., Lu, Z., and Li, S. (2021). Transductive few-shot classification on the oblique manifold. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8412–8422.

[155] Qu, Q., Sun, J., and Wright, J. (2014). Finding a sparse vector in a subspace: Linear sparsity using alternating directions. In *Advances in Neural Information Processing Systems*, volume 27.

[156] Qu, Q., Zhu, Z., Li, X., Tsakiris, M. C., Wright, J., and Vidal, R. (2020). Finding the sparsest vectors in a subspace: Theory, algorithms, and applications. arXiv:2001.06970.

[157] Ring, W. and Wirth, B. (2012). Optimization methods on Riemannian manifolds and their application to shape space. *SIAM Journal on Optimization*, 22(2):596–627.

[158] Rockafellar, R. T. and Wets, R. J.-B. (1998). *Variational Analysis*, volume 317. Springer Science & Business Media.

[159] Rozložník, M. (2018). *Saddle-point Problems and Their Iterative Solution*. Springer.

[160] Rudin, W. et al. (1976). *Principles of Mathematical Analysis*. McGraw-hill, 3 edition.

[161] Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*. SIAM.

[162] Sato, H. (2014). Riemannian conjugate gradient method for complex singular value decomposition problem. In *53rd IEEE Conference on Decision and Control*, pages 5849–5854. IEEE.

[163] Sato, H. (2016). A Dai-Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions. *Computational Optimization and Applications*, 64:101–118.

[164] Sato, H. (2017). Riemannian Newton-type methods for joint diagonalization on the Stiefel manifold with application to independent component analysis. *Optimization*, 66(12):2211–2231.

[165] Sato, H. (2021). *Riemannian Optimization and Its Applications*. SpringerBriefs in Electrical and Computer Engineering. Springer Cham.

[166] Sato, H. (2022). Riemannian conjugate gradient methods: General framework and specific algorithms with convergence analyses. *SIAM Journal on Optimization*, 32(4):2690–2717.

[167] Sato, H. and Iwai, T. (2013). A Riemannian optimization approach to the matrix singular value decomposition. *SIAM Journal on Optimization*, 23(1):188–212.

[168] Sato, H. and Iwai, T. (2015). A new, globally convergent Riemannian conjugate gradient method. *Optimization*, 64(4):1011–1031.

[169] Sato, H., Kasai, H., and Mishra, B. (2019). Riemannian stochastic variance reduced gradient algorithm with retraction and vector transport. *SIAM Journal on Optimization*, 29(2):1444–1472.

[170] Sato, H. and Sato, K. (2015). Riemannian trust-region methods for h^2 optimal model reduction. In *2015 54th IEEE Conference on Decision and Control (CDC)*, pages 4648–4655. IEEE.

[171] Sato, H. and Sato, K. (2017). Riemannian optimal system identification algorithm for linear MIMO systems. *IEEE Control Systems Letters*, 1(2):376–381.

[172] Sato, K. (2019). Riemannian optimal model reduction of stable linear systems. *IEEE Access*, 7:14689–14698.

[173] Sato, K. and Sato, H. (2018). Structure-preserving h^2 optimal model reduction based on the Riemannian trust-region method. *IEEE Transactions on Automatic Control*, 63(2):505–512.

[174] Schiela, A. and Ortiz, J. (2021). An SQP method for equality constrained optimization on Hilbert manifolds. *SIAM Journal on Optimization*, 31(3):2255–2284.

[175] Séguin, A. and Kressner, D. (2022). Continuation methods for Riemannian optimization. *SIAM Journal on Optimization*, 32(2):1069–1093.

[176] Selig, J. M. (2005). *Geometric Fundamentals of Robotics*. Springer.

[177] Shi, D., Gao, J., Hong, X., Boris Choy, S., and Wang, Z. (2021). Coupling matrix manifolds assisted optimization for optimal transport problems. *Machine Learning*, 110:533–558.

[178] Shilon, O. (2023). Randorthmat. MATLAB Central File Exchange. Retrieved September 9, 2023.

[179] Sikirić, M. D., Schürmann, A., and Vallentin, F. (2021). A simplex algorithm for rational cp-factorization. *Mathematical Programming*, 187:25–45.

[180] Smirnov, O. (2021). TensorFlow RiemOpt: A library for optimization on Riemannian manifolds. arXiv:2105.13921.

[181] So, W. and Xu, C. (2015). A simple sufficient condition for complete positivity. *Operators and Matrices*, 9(1):233–239.

[182] Song, G.-J. and Ng, M. K. (2020). Nonnegative low rank matrix approximation for nonnegative matrices. *Applied Mathematics Letters*, 105:106300.

[183] Sponsel, J. and Dür, M. (2014). Factorization and cutting planes for completely positive matrices by copositive projection. *Mathematical Programming*, 143:211–229.

[184] Sra, S. and Hosseini, R. (2013). Geometric optimisation on positive definite matrices for elliptically contoured distributions. In *Advances in Neural Information Processing Systems*, volume 26.

[185] Srivastava, A., Klassen, E., Joshi, S. H., and Jermyn, I. H. (2011). Shape analysis of elastic curves in euclidean spaces. *IEEE transactions on pattern analysis and machine intelligence*, 33(7):1415–1428.

[186] Steinlechner, M. (2016). Riemannian optimization for high-dimensional tensor completion. *SIAM Journal on Scientific Computing*, 38(5):S461–S484.

[187] Sun, J., Qu, Q., and Wright, J. (2017). Complete dictionary recovery over the sphere II: Recovery by Riemannian trust-region method. *IEEE Transactions on Information Theory*, 63(2):885–914.

[188] Sutherland, W. A. (2009). *Introduction to Metric and Topological Spaces*. Oxford University Press.

[189] Thanwerdas, Y. and Pennec, X. (2023). $\mathrm{O}(n)$-invariant Riemannian metrics on SPD matrices. *Linear Algebra and Its Applications*, 661:163–201.

[190] Theis, F. J., Cason, T. P., and Absil, P. A. (2009). Soft dimension reduction for ICA by joint diagonalization on the Stiefel manifold. In *Independent Component Analysis and Signal Separation: 8th International Conference, ICA 2009, Paraty, Brazil, March 15-18, 2009. Proceedings 8*, pages 354–361. Springer.

[191] Townsend, J., Koep, N., and Weichwald, S. (2016). Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation. *Journal of Machine Learning Research*, 17(137):1–5.

[192] Trendafilov, N. and Gallo, M. (2021). *Multivariate Data Analysis on Matrix Manifolds*. Springer, 1 edition.

[193] Tu, L. W. (2011). *An Introduction to Manifolds*. Springer New York, 2 edition.

[194] Udriste, C. (1994). *Convex Functions and Optimization Methods on Riemannian manifolds*. Springer Science & Business Media.

[195] Ungar, A. (2022). *A Gyrovector Space Approach to Hyperbolic Heometry*. Springer Nature.

[196] Ungar, A. A. (2008). *Analytic Hyperbolic Geometry and Albert Einstein's Special Theory of Relativity*. World Scientific.

[197] Utpala, S., Han, A., Jawanpuria, P., and Mishra, B. (2022). Rieoptax: Riemannian optimization in JAX. In *OPT 2022: Optimization for Machine Learning (NeurIPS 2022 Workshop)*.

[198] Vandereycken, B. (2013). Low-rank matrix completion by Riemannian optimization. *SIAM Journal on Optimization*, 23(2):1214–1236.

[199] Wächter, A. and Biegler, L. T. (2000). Failure of global convergence for a class of interior point methods for nonlinear programming. *Mathematical Programming*, 88:565–574.

[200] Wang, P., Liu, H., and So, A. M.-C. (2019). Globally convergent accelerated proximal alternating maximization method for l_1-principal component analysis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8147–8151. IEEE.

[201] Wen, Z. and Yin, W. (2013). A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142:397–434.

[202] Wright, S. J. (1997). *Primal-dual Interior-point Methods*. SIAM, Philadelphia.

[203] Wu, X., Wen, Z., and Bao, W. (2017). A regularized Newton method for computing ground states of Bose-Einstein condensates. *Journal of Scientific Computing*, 73:303–329.

[204] Xiao, N., Hu, X., Liu, X., and Toh, K.-C. (2022). CDOpt: A python package for a class of Riemannian optimization. arXiv:2212.02698.

[205] Yabe, H. and Yamashita, H. (1997). Q-superlinear convergence of primal-dual interior point quasi-Newton methods for constrained optimization. *Journal of the Operations Research Society of Japan*, 40(3):415–436.

[206] Yamada, I. and Ezaki, T. (2003). An orthogonal matrix optimization by dual Cayley parametrization technique. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pages 35–40.

[207] Yamakawa, Y. and Sato, H. (2022). Sequential optimality conditions for nonlinear optimization on Riemannian manifolds and a globally convergent augmented Lagrangian method. *Computational Optimization and Applications*, 81:1–25.

[208] Yamashita, H. and Yabe, H. (1996). Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization. *Mathematical Programming*, 75:377–397.

[209] Yang, W. H., Zhang, L.-H., and Song, R. (2014). Optimality conditions for the nonlinear programming problems on Riemannian manifolds. *Pacific Journal of Optimization*, 10(2):415–434.

[210] Yang, Z. and Oja, E. (2010). Linear and nonlinear projective nonnegative matrix factorization. *IEEE Transactions on Neural Networks*, 21(5):734–749.

[211] Yao, T.-T., Bai, Z.-J., Jin, X.-Q., and Zhao, Z. (2020). A geometric Gauss-Newton method for least squares inverse eigenvalue problems. *BIT Numerical Mathematics*, 60:825–852.

[212] Yao, T.-T., Bai, Z.-J., Zhao, Z., and Ching, W.-K. (2016). A Riemannian Fletcher-Reeves conjugate gradient method for doubly stochastic inverse eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 37(1):215–234.

[213] Ye, Y. (1997). *Interior Point Algorithms: Theory and Analysis*. John Wiley & Sons.

[214] Zhang, C., Chen, X., and Ma, S. (2023). A Riemannian smoothing steepest descent method for non-Lipschitz optimization on embedded submanifolds of r^n. *Mathematics of Operations Research*.

[215] Zhang, H., J Reddi, S., and Sra, S. (2016). Riemannian SVRG: Fast stochastic optimization on Riemannian manifolds. In *Advances in Neural Information Processing Systems*, volume 29.

[216] Zhang, T., Wiesel, A., and Greco, M. S. (2013). Multivariate generalized Gaussian distribution: Convexity and graphical models. *IEEE Transactions on Signal Processing*, 61(16):4141–4148.

[217] Zhao, Z., Bai, Z.-J., and Jin, X.-Q. (2015). A Riemannian Newton algorithm for nonlinear eigenvalue problems. *SIAM Journal on Matrix Analysis and Applications*, 36(2):752–774.

[218] Zheng, L. and Tse, D. N. C. (2002). Communication on the Grassmann manifold: A geometric approach to the noncoherent multiple-antenna channel. *IEEE transactions on Information Theory*, 48(2):359–383.

[219] Zhu, X. (2017). A Riemannian conjugate gradient method for optimization on the Stiefel manifold. *Computational Optimization and Applications*, 67:73–110.

[220] Zhu, X. and Sato, H. (2020). Riemannian conjugate gradient methods with inverse retraction. *Computational Optimization and Applications*, 77:779–810.

List of Author's Papers Related to This Thesis

1. LAI ZHIJIAN and Akiko Yoshise, "Completely Positive Factorization by a Riemannian Smoothing Method", *Computational Optimization and Applications*, Vol.83, No.3, 2022, pp.933-966.
2. LAI ZHIJIAN and Akiko Yoshise, "Riemannian Interior Point Methods for Constrained Optimization on Manifolds", *Journal of Optimization Theory and Applications*, to appear.

Biographical Sketch

Zhijian Lai, the son of Liangao Lai and Fendi Dai, was born on February 11, 1995, in Puyang, Henan province of China. He has always been supported and inspired by his family, including his older brother, Qiang Lai. He began his academic journey at Dongbei University of Finance and Economics in Dalian, China, where he received his Bachelor's Degree in Management, specializing in Logistics Management from 2013 to 2017.

Seeking to broaden his horizons, Zhijian moved to Japan, where he attended the ARC Tokyo Japanese Language School from 2017 to 2019, focusing on graduate school preparation. This experience paved the way for his entry into the University of Tsukuba. At Tsukuba, he completed his Master's degree in Policy and Planning Sciences from 2019 to 2021, conducting research under the supervision of Prof. Akiko Yoshise. His master's thesis was titled "A New Method for Completely Positive Matrix Factorization." Continuing at the University of Tsukuba, Zhijian is currently pursuing his Ph.D. in Policy and Planning Sciences, expected to be completed by March 2024. His Ph.D. thesis, also supervised by Prof. Yoshise, is titled "Riemannian Optimization Algorithms for Applications and Their Theoretical Properties." Zhijian's research topics include nonlinear optimization on Riemannian manifolds. He made significant contributions to extend the classical primal-dual interior point method to Riemannian manifolds.

Looking ahead, Zhijian Lai is about to launch a new chapter in his career. Starting from May 2024, he will join Prof. Zaiwen Wen's research group at the Beijing International Center for Mathematical Research (BICMR) at Peking University as a postdoctoral fellow. He will continue his research in mathematical optimization.

February, 2024