Convexity-like Structures in Linear Algebra and Applications in Algorithmic Computation

Alimisis, Foivos

UNIVERSITÉ DE GENÈVE
Section de mathématiques

FACULTÉ DES SCIENCES
Prof. Bart Vandereycken

Convexity-like Structures in Linear Algebra and Applications in Algorithmic Computation

Ph.D. THESIS

presented to the Faculty of Science, University of Geneva
for obtaining the degree of Doctor of Mathematics.

by

Foivos Alimisis

from

Lamia (Greece)

PhD N° 5886

Atelier de reproduction de la Section de physique
Geneva, 2024

UNIVERSITÉ DE GENÈVE

FACULTÉ DES SCIENCES

DOCTORAT ÈS SCIENCES, MENTION MATHEMATIQUES

Thèse de Monsieur Foivos ALIMISIS

intitulée :

«Convexity-like Structures in Linear Algebra and Applications in Algorithmic Computation»

La Faculté des sciences, sur le préavis de

Monsieur B. VANDEREYCKEN, professeur associé et directeur de thèse
Section de mathématiques

Monsieur G. VILMART, docteur
Section de mathématiques

Monsieur N. BOUMAL, professeur assistant
Institute of Mathematics, EPFL, Lausanne

Monsieur Y. SAAD, professeur
Department of Computer Science & Engineering, University of Minnesota, Minnesota, United States

autorise l'impression de la présente thèse, sans exprimer d'opinion sur les propositions qui y sont énoncées.

Genève, le 17 février 2025

Thèse - 5886 -

La Doyenne

N.B. - La thèse doit porter la déclaration précédente et remplir les conditions énumérées dans les "Informations relatives aux thèses de doctorat à l'Université de Genève".

Abstract

This thesis primarily addresses the following problem: why can certain non-convex optimization problems be solved efficiently? We focus on two important problems in linear algebra and uncover a convexity-like structure that may answer the aforementioned question. This convexity-like structure does not hold in a Euclidean space but rather "geodesically" on a Riemannian manifold. Thus, the main components of this thesis are linear algebra, optimization, and differential geometry. Since it is challenging for a reader to be familiar with all these prerequisites, we strive to present each topic as independently as possible.

The structure analyzed in this thesis facilitates numerous applications in the field of numerical linear algebra. Consequently, a significant portion of the thesis is dedicated to these applications. They include eigenvalue problems in a distributed setting, a new analysis for preconditioned eigenvalue solvers, as well as the development and analysis of new eigenvalue solvers for the most elementary cases. This should certainly be of interest to readers with a background in numerical linear algebra.

On the other hand, readers with an optimization background may view this thesis as an illustration of the importance of the aforementioned structure (weak-quasi-strong convexity). Insights into such structures have appeared not only in linear algebra but also in deep learning. In the final section, we show that this structure is indeed special for optimization in general, as it is in some sense necessary for linear convergence of gradient descent with respect to the error defined by the distances of iterates to the set of optima.

Readers interested in differential geometry may see this thesis as a good example of the power of optimization on Riemannian manifolds. Although differential geometry is not the central focus of this thesis, all the structures discussed hold over curved spaces. This serves as an excellent example of the primary goal of optimization on Riemannian manifolds: solving typically non-convex problems within geometries where they satisfy a certain convexity structure.

Résumé

Cette thèse traite principalement du problème suivant: pourquoi certains problèmes d'optimisation non-convexes peuvent être résolus rapidement. Nous nous concentrons sur deux problèmes importants d'algèbre linéaire et révélons une structure de type convexité qui peut répondre à la question précédente. Cette structure de type convexité n'est pas valable dans un espace euclidien, mais plutôt de manière "géodésique" sur une variété riemannienne. Ainsi, les principales composantes de cette thèse sont l'algèbre linéaire, l'optimisation et la géométrie différentielle. Comme il est difficile pour un lecteur de connaître tous ces prérequis, nous nous efforçons de présenter chaque sujet aussi indépendamment que possible.

La structure analysée dans cette thèse permet de faciliter de nombreuses applications dans le domaine de l'algèbre linéaire numérique. Ainsi, une grande partie de la thèse leur est consacrée. Elles incluent les problèmes de valeurs propres dans un régime distribué, une nouvelle analyse pour les solveurs de valeurs propres préconditionnés, mais aussi le développement et l'analyse de nouveaux solveurs de valeurs propres dans les cas les plus élémentaires. Ceci devrait certainement intéresser le lecteur issu de l'algèbre linéaire numérique.

D'un autre côté, le lecteur qui est plus proche de l'optimisation, peut voir cette thèse comme une illustration de l'importance de la structure mentionnée ci-dessus (weak-quasi-strong convexity). Des aperçus de telles structures sont apparus non seulement en algèbre linéaire, mais aussi en apprentissage profond (deep learning). Dans la dernière section, nous montrons que cette structure est en effet spéciale pour l'optimisation en général, car elle est en quelque sorte nécessaire pour la convergence linéaire de la descente de gradient en ce qui concerne les distances des itérés à l'ensemble des optima.

Le lecteur intéressé par la géométrie différentielle peut voir dans cette thèse un bon exemple de la puissance de l'optimisation sur les variétés riemanniennes. Bien que la géométrie différentielle ne soit pas le centre de cette thèse, toutes les structures discutées sont valables sur des espaces courbes. C'est un bon exemple de l'objectif principal de l'optimisation sur les variétés riemanniennes, c'est-à-dire résoudre des problèmes généralement non-convexes dans des géométries où ils satisfont une certaine structure de convexité.

Preface

The years I spent in the university of Geneva pursuing this PhD work have been extremely productive. Most of the time, I felt doing something meaningful, which is not to be taken for granted. In the meantime, I met great people.

The most important person contributing to my academic success has been my PhD advisor Bart Vandereycken. He has created for us a healthy and drama-free research environment, with good guidance but also plenty of personal freedom to pursue our own directions. Again, these are not to be taken for granted. His personal success story (inside and outside academia) makes him also a great role model for any young researcher. Bart, I owe you really a lot!

Besides Bart, I have been fortunate to be advised by other great people throughout my academic career. These include my MSc advisor Aurelien Lucchi, who transformed me from an ignorant student to a promising junior researcher after countless long meetings and personal effort. He also gave me the next big opportunity of my life by hiring me in his newly-established group in the university of Basel. Aurelien, I am looking forward to working with you again!

Another great professor, who I was fortunate to be advised by is Dan Alistarh. Dan hired me as an intern in IST Austria and has shown his interest in me in numerous ways. I really appreciate that. Actually, the cornerstone ideas behind this PhD thesis were born when working in his group. We are still in touch and I feel fortunate about that.

I am also certainly grateful to my large group of collaborators during my PhD and pre-PhD years. This is consisted by Antonio Orvieto, Gary Becigneul, Peter Davies, Simon Vary, Pierre-Antoine Absil, Yousef Saad, Nian Shao and Daniel Kressner. A large part of this work would not be possible without you; Special thanks to my PhD committee consisted by Nicolas Boumal, Yousef Saad and Gilles Vilmart. It is always important to have top experts reviewing one's work.

My years at the university of Geneva have been marked by the interaction with many great colleagues and external friends, who are too many to list. Special thanks to my great office mates, who were always a pleasure to greet and talk to every day.

A big thank you goes also to my parents, who are great people, and taught me about the value of education, hard work and responsibility. If it was not my dad teaching me math from an early age, perhaps I would never have pursued this career path. My parents also spent a large part of their savings for my MSc studies in Zurich, while maintaining a simple life, which I find remarkable. Part of my studies in Zurich was funded also by money that my now deceased grandma concentrated for that reason. The vision of this hardworking and poor woman with low education level and tough life is truly inspiring.

Last but not least, I would like to thank my girlfriend Ran, who is the most

loving person I have ever met. She is a great, smart and talented woman. Over the last two years, she gave me the emotional tranquility that allowed me to work hard on my goals. She has given a deeper meaning in my life and her presence has helped me to put my future goals into a deeper perspective.

Foivos Alimisis

Contents

1 Introduction

Tractability in optimization has long been associated with convexity. The field of convex optimization has systematically studied algorithms and their convergence guarantees for convex optimization problems with excellent results [80]. Unfortunately, convexity turns out to be an unrealistic scenario for many problems of interest. This category includes problems from basic linear algebra to advanced deep learning models. Moreover, many of these problems turn out to be tractable, i.e. their non-convex structure is benign in some sense. Such a phenomenon can be observed experimentally (for instance it has been repeatedly observed that stochastic gradient descent optimizes over-parametrized deep neural networks fast and accurately) or even theoretically as in the case of certain linear algebra applications. An example for the latter is the symmetric eigenvalue problem, which, while non-convex, it is long known to be solvable easily by many algorithms that have been developed by the numerical linear algebra community [96], the most popular of them being the power method.

In this thesis, we reveal a convexity-like structure for two of the most popular problems in linear algebra, namely the symmetric eigenvalue problem and the problem of polar decomposition. We also study numerous applications of this theory to practical algorithmic design and analysis, improving the state-of-the-art in numerical linear algebra using off-the-self techniques from convex optimization. In Section 8, we study the importance of the aforementioned convexity-like structures for optimization in general, with a few surprising results.

One aspect that co-exists on the side of this work is the importance of optimization over Riemannian manifolds. The problems that we deal with can be naturally posed on Riemannian manifolds and there are good reasons to do so. The convexity-like structures that we study do not hold over some Euclidean space, but rather "geodesically" over some Riemannian manifold. As Riemannian adaptations of popular Euclidean algorithms are well-studied under certain function classes (see for instance [54, 112, 118, 119]), more effort should be invested in identifying problems, where a change in geometry of the search space can yield to the rise of such function classes. Our results show that the symmetric eigenvalue problem and polar decomposition offer great examples.

We continue our introduction by discussing basic concepts of the three fields that intersect in this thesis: optimization, linear algebra and differential geometry.

1.1 Basics from optimization

Optimization is one of the most vibrant fields in applied mathematics. Its success can be largely explained by its importance in training machine learning

models. However, optimization problems can be found in many other fields of mathematics and applied sciences, including linear algebra. Many excellent textbooks exist for the interested reader, see for instance [22, 80, 87].

Considering the problem of minimizing a function $f : \mathbb{R}^n \to \mathbb{R}$ that attains a minimum

$$\min_{x \in \mathbb{R}^n} f(x), \tag{1.1.0.1}$$

the most popular algorithm to deal with it is *gradient descent*, which dates back to the work of Cauchy [68].

This extremely simple algorithm assumes that f is differentiable, thus its gradient can be computed at any point. It takes the form:

$$x_{t+1} = x_t - \eta \nabla f(x_t), \tag{1.1.0.2}$$

where x_{t+1} is a new guess for a minimizer of f starting from a previous guess x_t and $\eta > 0$ is a step size. The first important property which guarantees that gradient descent behaves reasonably well is the Lipschitz continuity of the gradient:

Definition 1.1 (L-smoothness) *A function $f : \mathbb{R}^n \to \mathbb{R}$ is called L-smooth if its gradient is L-Lipschitz continuous, i.e.*

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|,$$

for all $x, y \in \mathbb{R}^n$.

L-smoothness implies a number of interesting properties (see [120]):

Proposition 1.2 *If f is L-smooth, then*

- *The largest eigenvalue of its Hessian is uniformly upper bounded by L in absolute value.*
- *For all $x, y \in \mathbb{R}^n$, it holds*

$$f(y) - f(x) \leq \langle \nabla f(x), y - x \rangle + \frac{L}{2}\|y - x\|^2.$$

- *For all $x \in \mathbb{R}^n$, it holds*

$$f(x) - f^* \geq \frac{1}{2L}\|\nabla f(x)\|^2,$$

  where f^* is the minimum of f.

Under L-smoothness, gradient descent can be guaranteed to converge to some critical point of f, i.e. a point x^*, such that $\nabla f(x^*) = 0$.

Under extra convexity-like assumptions one can show stronger convergence guarantees about gradient descent:

Proposition 1.3 *If a function f is convex, then all local minima are global. If in addition it is L-smooth, then gradient descent* (1.1.0.2) *with step size $\eta = 1/L$ converges to the global minimum f^* with an* algebraic *convergence rate:*

$$f(x_t) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{t+1},$$

where $x_0 \in \mathbb{R}^n$ is the starting point of the iteration and $x^* \in \mathbb{R}^n$ some of the global optima.

Proof The proof that every local minimum is global can be found in Proposition 1.2 in [26] and the convergence rate in Theorem 3.3 in [26]. ■

Proposition 1.4 *If a function f is μ-strongly convex, then the global minimizer is unique. If, in addition, f is L-smooth, then gradient descent* (1.1.0.2) *with step size $\eta = 1/L$ converges to the global minimizer (let it be x^*) with a* linear *convergence rate:*

$$\|x_t - x^*\|^2 \leq \left(1 - \frac{\mu}{L}\right)^t \|x_0 - x^*\|^2.$$

Proof See Theorem 3.10 in [26]. ■

Notice that the previous convergence rate is with respect to the distances of the iterates to the optimum. One can also show a convergence rate with respect to the values of the function f to the minimum f^*:

Proposition 1.5 *For a μ-strongly convex and L-smooth function f, the iterates of gradient descent* (1.1.0.2) *with $\eta = 1/L$ satisfy*

$$f(x_t) - f^* \leq \left(1 - \frac{\mu}{L}\right)^t (f(x_0) - f^*).$$

Proof Given the simplicity and the instructive nature of this proof, we present it in detail.

Since f is L-smooth, we have that (see [120], Lemma 4)

$$f(x_{t+1}) \leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2.$$

Since $x_{t+1} - x_t = \frac{1}{L}\nabla f(x_t)$, we have

$$f(x_{t+1}) - f^* \leq f(x_t) - f^* - \frac{1}{2L}\|\nabla f(x)\|^2.$$

Since f is μ-strongly convex, we have that (see Lemma 3(i) in [120])

$$\|\nabla f(x)\|^2 \geq 2\mu(f(x) - f^*), \text{ for all } x. \tag{1.1.0.3}$$

Writing this inequality for $x = x_t$ and combining with the previous inequality, we get that

$$f(x_{t+1}) - f^* \leq f(x_t) - f^* - \frac{\mu}{L}(f(x_t) - f^*) = \left(1 - \frac{\mu}{L}\right)(f(x_t) - f^*).$$

Extending this inequality by induction, we get the desired result.

■

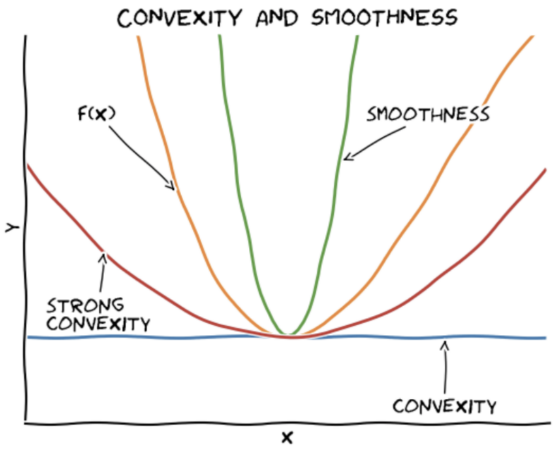


Figure 1.1: A joint illustration of the notions of (strong-)convexity and smoothness. Such function cannot grow faster than any quadratic and cannot shrink faster than a line in the convex case or a quadratic in the strongly convex one.

A close inspection in the proof of Proposition 1.5 reveals that one does not need strong convexity, but only its weaker implication (1.1.0.3). This was first observed by Boris Polyak [93] and, as Stanisław Łojasiewicz was simultaneously studying more general version of the condition, it took the name Polyak-Łojasiewicz (PL) condition.

The PL condition is much more general than strong convexity as it includes non-convex optimization problems. Interesting examples of problems that satisfy a PL condition, but are not strongly convex, include logistic regression (see [56], section 2.3) and certain architectures of (usually overparametrized) deep neural networks (see [105], Lemma 7.12). An important result coming from [56] shows that among all properties that the optimization community has come up with (until that point) in order to guarantee a linear convergence rate for gradient descent, PL is the weakest. Even deeper, [1] shows (Theorem 5) that PL is a *necessary* condition for gradient descent to have linear convergence with respect to function values to the minimum, when applied to a function with Lipschitz continuous gradient.

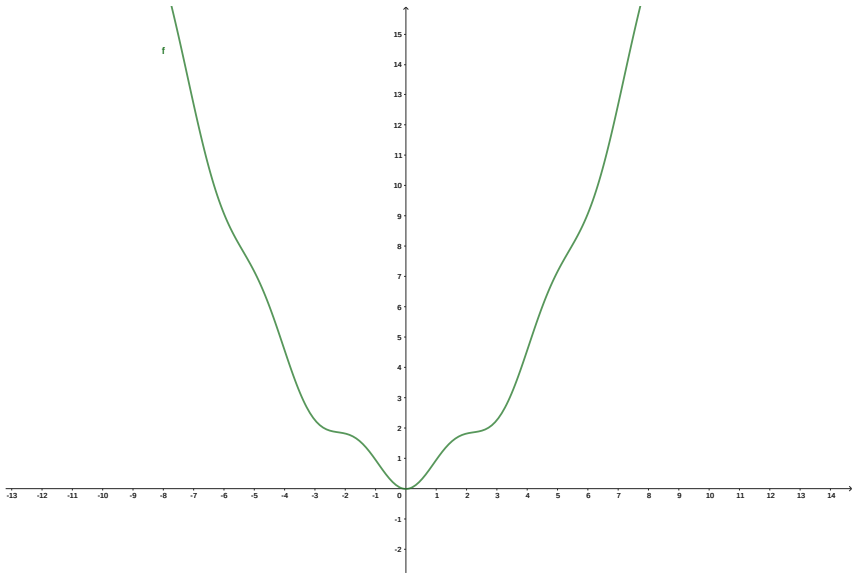


Figure 1.2: A classic example of a function which satisfies a PL condition but is not convex.

While the PL condition is perhaps the most popular non-convexity property studied in the realm of convex optimization, it will turn out to be insufficient for our problems of interest. A stronger property than PL that facilitates our future analysis is the following:

Definition 1.6 (Weak-quasi-strong convexity) *A function $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ is called (a, μ)-weak-quasi-strongly convex (WQSC) in a set $E \subseteq \mathbb{R}^n$, if it has a unique optimum x^* in E and for all $x \in E$ we have*

$$f(x) - f^* \leq \frac{1}{a}\langle \text{grad} f(x), x - x^* \rangle - \frac{\mu}{2}\|x - x^*\|^2,$$

for some constants $a, \mu > 0$.

Remark 1.1 *WQSC is equivalent with the so-called weak-quasi convexity property [40] and the more well-known quadratric growth property [33] holding simultaneously.*

Note that (a, μ)-WQSC includes the class of μ-strongly convex functions for $a = 1$. Even when $a \neq 1$, it guarantees a PL condition:

Proposition 1.7 *If f is (a, μ)-WQSC in E, then it satisfies the PL condition*

$$\|\nabla f(x)\|^2 \geq 2a^2\mu(f(x) - f^*),$$

for all $x \in E$.

Proof The proof is simple and can be found in [25] (Lemma 3.2). ■

What makes weak-quasi-strong convexity interesting is that it guarantees linear convergence of gradient descent (1.1.0.2) *with respect to distances of the iterates to the optimum*, in contrast with the PL condition, which guarantees convergence only with respect to function values.

Proposition 1.8 *If f is L-smooth and (a, μ)-WQSC in $\mathbb{R}^n$, then gradient descent with step size $\eta = a/L$ produces iterates that satisfy*

$$\|x_t - x^*\|^2 \leq \left(1 - a^2 \frac{\mu}{L}\right)^t \|x_0 - x^*\|^2.$$

Proof The proof can be found in Lemma 4.2 of [25]. ■

Remark 1.2 *We make the choice to not bother too much with the proofs of these results in our introduction as they have already appeared in related works. Later in the text, we will need versions of these results in slightly different settings, like in a Riemannian regime, or with a more general step size, or in a more restrictive domain etc. In these cases, we will revisit the proofs of these results in detail.*

Similarly with PL, we will show that weak-quasi-strong convexity has a special meaning in optimization, namely, except sufficient, it is also necessary for linear convergence of gradient descent with respect to distances of the iterates to the optimum. This is the content of Section 8.

Except gradient descent, another popular algorithm that will concern us is an accelerated version of gradient descent with momentum in the style proposed by Yurii Nesterov in his seminal work [79]. This algorithm can take the simple form of Algorithm 1.1 presented in page 78 of [80].

Algorithm 1.1 Accelerated gradient descent with Nesterov momentum

1: Choose $x_0 \in \mathbb{R}^n$ and set $v_0 = x_0$.
2: **for** $t \geq 0$ **do**
3: Compute $\alpha_t > 0$ such that $\alpha_t^2 = \frac{(1-\alpha_t)\gamma_t + \alpha_t \mu}{L}$.
4: Set $y_t = \frac{\alpha_t \gamma_t v_t + \gamma_{t+1} x_t}{\gamma_t + \alpha_t \mu}$.
5: Set $x_{t+1} = x_t - \frac{1}{L}\nabla f(y_t)$.
6: Set $v_{t+1} = \frac{1}{\gamma_{t+1}}((1-\alpha_t)\gamma_t v_t + \alpha_t \mu y_t - \alpha_t \nabla f(y_t)$.
7: **end for**

Accelerated gradient descent is more complicated than gradient descent, but still constructed by simple ideas. The original convergence analysis is made for convex or strongly convex functions using a technique called "estimate

sequence". In the strongly convex case (which is more of interest for us), such algorithm produces iterates that satisfy

$$f(x_t) - f^* \leq \left(1 - \sqrt{\frac{\mu}{L}}\right)^t (f(x_0) - f^*).$$

The reader can refer to Theorem 2.2.3 in [80]. The "acceleration" is reflected on the square root that appears around the inverse condition number μ/L. Such number can be extremely small (close to 0) in many practical applications, in which case its square root is substantially larger. This algorithm is not the most practical, as there are certain hyperparameters that need to be set accurately in order to achieve the desired convergence rate. It has though high theoretical value as it is in some sense "optimal" among all first-order methods (i.e. methods that access only function values and gradients).

In our case, we will not be dealing with strongly convex problems. However, even weak-quasi-strong convexity is enough to design and analyse an estimate sequence that gives rise to an algorithm with an accelerated convergence rate. This will be important in Section 6.

1.2 Basics from linear algebra

1.2.1 The symmetric eigenvalue problem

One of the main problems presented in this thesis is the computation of some eigenvalues and associated eigenvectors of some symmetric matrix $A \in \mathbb{R}^{n\times n}$. Together with the algorithmic solution of linear systems, eigenvalue problems have been prototypical for the domain of numerical linear algebra. They have been prototypical for the field of optimization over Riemannian manifolds as well. For instance, it is the most standard problem treated in the popular textbook [3], while it appears also in earlier efforts, see for instance [17]. The latter is a good example of an effort from the Riemannian optimization community to design more competitive (trust-region-style) methods for the symmetric eigenvalue problem, using some novel machinery. These early attempts are focused mostly on computational aspects of useful Riemannian quantities and not so much on strong convergence guarantees.

In this section, we present some basics related to eigenvalues and eigenvectors and refer the reader to classic textbooks [39, 96, 110] for more.

An eigenvalue $\lambda \in \mathbb{R}$ and an associated eigenvector $v \in \mathbb{R}^n \setminus \{0\}$ of a symmetric matrix $A \in \mathbb{R}^{n\times n}$ is a pair such that

$$Ax = \lambda x.$$

Eigenvalues and eigenvectors are well-defined for a larger class of matrices, but we shall stick to the case of symmetric matrices, as then all eigenvalues are guaranteed to be real numbers. One can write a symmetric eigenvalue

problem also in matrix form. To that end, we denote by $\Lambda \in \mathbb{R}^{n\times n}$ a diagonal matrix containing some eigenvalues of A in its diagonal entries, while $V \in \mathbb{R}^{n\times k}$ denotes a matrix featuring k-many eigenvectors in its columns. The desired relationship now takes the form

$$AV = V\Lambda. \tag{1.2.1.1}$$

From now on, we will usually be dealing with a multiple eigenvalue-eigenvector problem in matrix form. One question that we tackle is computing the largest k eigenvalues and associated eigenvectors of the matrix A. Let us denote the eigenvalues of A in decreasing order as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_k \geq \lambda_{k+1} \geq \ldots \geq \lambda_n$. The first k eigenvalues are the wanted ones, while the last $n-k$ are unwanted. We also denote by $\delta = \lambda_k - \lambda_{k+1}$ the gap between the wanted and unwanted eigenvalues. Depicting this situation in matrix form requires to store the wanted and unwanted eigenvalues separately in the diagonal entries of two diagonal matrices $\Lambda_\alpha = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$ and $\Lambda_\beta = \mathrm{diag}(\lambda_{k+1}, \ldots, \lambda_n)$. We can also define a matrix $V_\alpha = \begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix}$, such that $V_\alpha^T V_\alpha = I_k$, that contains the eigenvectors corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_k$ and $V_\beta = \begin{bmatrix} v_{k+1} & \cdots & v_n \end{bmatrix}$, such that $V_\beta^T V_\beta = I_{n-k}$ and $V_\alpha^T V_\beta = 0_{k\times(n-k)}$, that contains the eigenvectors corresponding to the eigenvalues $\lambda_{k+1}, \ldots, \lambda_n$.

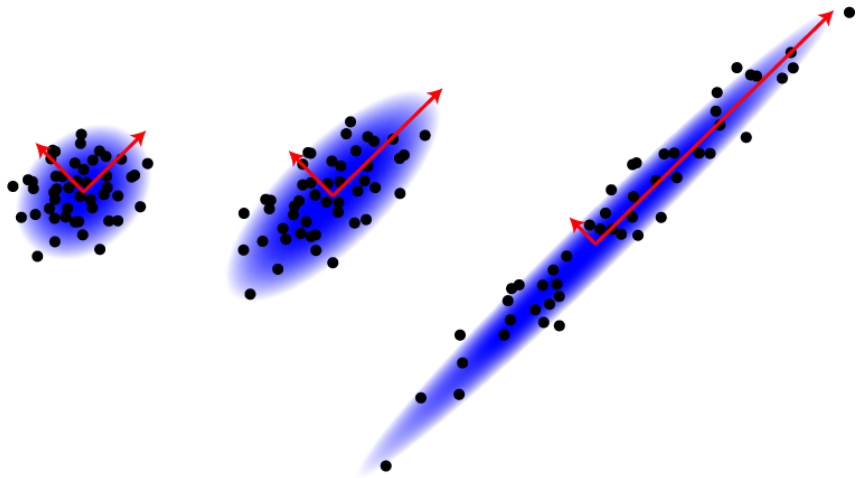


Figure 1.3: Illustration of some datasets and the eigenvectors of their covariance matrices. These eigenvectors indicate directions of maximum or minimum covariance. Picture by Jesse Johnson.

The most popular algorithm for solving the problem (1.2.1.1) is the so-called subspace iteration. Starting from an initial $n \times k$ matrix X_0, one updates as

$$X_{t+1} = \mathrm{QR}(AX_t). \tag{1.2.1.2}$$

$\mathrm{QR}(\cdot)$ means that the algorithm keeps the orthogonal factor of the QR decomposition of AX_t. This is done in order to prevent the columns of X_t to converge

to the same eigenvector corresponding to the largest eigenvalue.

This algorithm is very simple and comes with an extremely simple convergence analysis. For phrasing a convergence result, we need the notion of principal angles between subspaces.

Definition 1.9 (Principal angles) *Given two subspaces* $\mathrm{Span}(X), \mathrm{Span}(Y) \subseteq \mathbb{R}^n$ *of dimension* k *with* $X, Y \in \mathbb{R}^{n\times k}$ *orthonormal, the principal angles between them are* $\theta_1, \theta_2, \ldots, \theta_k \in [0, \pi/2]$*, if the SVD of* $Y^T X$ *can be written as*

$$Y^T X = U_1 \cos\theta \; V_1^T$$

where $U_1 \in \mathbb{R}^{k\times k}, V_1 \in \mathbb{R}^{k\times k}$ *are orthogonal and* $\cos\theta := \mathrm{diag}(\cos\theta_1, \ldots, \cos\theta_k)$.

Notice that the definition of principal angles is independent of the specific orthonormal matrices which represent the subspaces. This notion captures how far away two subspaces are, similarly with the notion of the angle between two vectors. Without loss of generality, we will treat the principal angles between two subspaces as ordered $\theta_1 \leq \theta_2 \leq \ldots \leq \theta_k$.

We now state a simple convergence result about subspace iteration (Theorem 8.2.1 in [39]):

Proposition 1.10 *Let* $X_0 \in \mathbb{R}^{n\times k}$ *be an orthonormal matrix, such that* $X_0^T V_\alpha$ *is non-singular. Let* X_t *be the iterates of subspace iteration* (1.2.1.2)*. Then, the largest principal angle* θ_k^t *between* X_t *and* V_α *satisfies*

$$\tan\theta_k^t \leq \left|\frac{\lambda_k}{\lambda_{k+1}}\right|^t \tan\theta_0.$$

Notice that if the spectral gap δ is strictly positive, then the previous result gives a linear convergence rate. If the spectral gap is 0, then it just states that the principal angles do not increase over the course of the algorithm.

Subspace iteration is a very handy algorithm, but it can suffer from poor performance, especially in the case that the spectral gap is tiny (this happens a lot in practical applications). The probably simplest idea on how to accelerate subspace iteration is filtering the objective matrix using some polynomial. A complete exposition of this process can be found in Chapter 7 of [96]. A general version of such algorithm reads as:

$$X_{t+1} = \mathrm{QR}(p_t(A)X_t), \tag{1.2.1.3}$$

where p_t is some polynomial of degree t. If $p_t(x) = x^t$, then we recover the case of vanilla subspace iteration. $\mathrm{QR}(\cdot)$ again keeps the orthogonal factor of the QR decomposition of $p_t(A)X_t$. For reasons that are beyond the scope of this thesis, the optimal choice for the polynomial p_t is

$$p_t(x) = C_t((x-c)/h) \text{ where } C_t \text{ is a Chebyshev polynomial of degree } t.$$

Here c and h are scaling parameters that depend on the (unknown in general) eigenvalues of A. This method accelerates over subspace iteration (1.2.1.2), but this acceleration comes at the cost of tuning the previously discussed hyperparameters which depend on unknown quantities. The convergence guarantees of this method with optimal filtering (also called Chebyshev iteration) can be found in Section 7.4.1 of [96].

Another family of "accelerated" algorithms is Krylov methods (Chapter 6 in [96]). Such algorithms are based on the construction of a Krylov subspace: starting from an orthonormal matrix $X \in \mathbb{R}^{n\times k}$, one constructs the subspace

$$\text{Span}\{X, AX, A^2X, ..., A^mX\}.$$

Methods that belong to this family build iteratively an orthogonal basis for such Krylov subspace. This orthogonal basis serves as a good approximation of the eigenvectors of the matrix A.

Krylov methods have similar convergence guarantees with Chebyshev iteration, without the messy selection of hyperparameters. The price that needs to be payed though is manifold: the cost of Krylov methods is not fixed per iteration, but rather increases. Another serious issue is that in certain applications (e.g. in electronic structure calculations [121]) the matrix A changes a little during the course of the algorithm. This makes Krylov methods unsuitable, as one needs to construct the whole Krylov subspace with the same matrix at once.

The final algorithm we would like to mention is, by some measures, an optimal eigenvalue solver. It is called locally optimal block preconditioned conjugate gradients (LOBPCG) algorithm and first appeared in seminal works by Andrew Knyazev [59, 60]. LOBPCG essentially computes the iterate that maximizes the Rayleigh quotient, chosen from a space constructed by the current iterate, a gradient-related term and a momentum term. It is at least as fast as the basic preconditioned eigenvalue solver (PINVIT) [61], but efforts to show that indeed accelerates over PINVIT have been proven illusive. That is to say that the excellent practical performance of LOBPCG is not backed by theoretical results in a solid way. Nevertheless, it is considered state-of-the-art from a practical point of view. This is why we test our algorithms of Sections 5 and 6 primarily against LOBPCG. Research on the theoretical guarantees of LOBPCG (and PINVIT) are still an active area of research, on which this thesis contributes (Section 4). Unfortunately, the analysis of Section 4 applies only to the basic PINVIT algorithm and not to the more advanced LOBPCG version, for reasons that are discussed there.

There are of course a lot more methods for solving large-scale eigenvalue problems, a complete exposition of them though would need the length of a textbook. Also, it would not be really helpful for the reader of this thesis. The traditional numerical linear algebra techniques have perhaps started to reach a historical limit and progress through them is really incremental. On

the other hand, there is some room for fresh approaches based on ideas of (geodesically) convex optimization on Riemannian manifolds. This strategy requires formulating the symmetric eigenvalue problem as an optimization problem.

A good point to start is by noticing that computing a set of largest eigenvalues of a matrix $A \in \mathbb{R}^{n\times n}$ can be formulated as the minimization of the function

$$f(X) = -\operatorname{Tr}(X^T A X)$$

over the set of $n \times k$ matrices with orthonormal columns. Indeed, from Fan's trace minimization theorem (see, e.g., [47, Corollary 4.3.39]) we know that

$$\min\{f(X)\colon X \in \mathbb{R}^{n\times k}, X^T X = I_k\} = -(\lambda_1 + \ldots + \lambda_k) = -\operatorname{Tr}(\Lambda_\alpha) =: f^*. \tag{1.2.1.4}$$

An optimum of this problem is the matrix $V_\alpha = \begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix}$ defined previously. If the spectral gap δ is strictly positive, then $\operatorname{Span}(V_\alpha)$ is unique; otherwise, we can choose any v_k from a subspace with dimension equal to the multiplicity of λ_k. It is readily seen that $f(V_\alpha) = -(\lambda_1 + \cdots + \lambda_k)$. In fact, all minimizers of (1.2.1.4) are of the form $V_\alpha Q$ with Q a $k \times k$ orthogonal matrix. We also define $V_\beta = \begin{bmatrix} v_{k+1} & \cdots & v_n \end{bmatrix}$ that contains the eigenvectors corresponding to the eigenvalues $\lambda_{k+1}, \ldots, \lambda_n$. Its columns span the orthogonal complement of $\operatorname{Span}(V_\alpha)$ in $\mathbb{R}^n$ and thus $V_\beta^T V_\beta = I_{n-k}$ and $V_\alpha^T V_\beta = 0_{k\times(n-k)}$.

Since $\operatorname{Span}(V_\alpha) = \operatorname{Span}(V_\alpha Q)$, it is more natural to consider this problem as a minimization problem on the Grassmann manifold $\operatorname{Gr}(n, k)$, i.e. the set of k-dimensional subspaces in $\mathbb{R}^n$. Let us therefore redefine the objective function as

$$f(\mathcal{X}) = -\operatorname{Tr}(X^T A X) \text{ where } \mathcal{X} = \operatorname{Span}(X) \text{ for } X \in \mathbb{R}^{n\times k} \text{ s.t. } X^T X = I_k. \tag{1.2.1.5}$$

This cost function can be seen as a block version of the standard Rayleigh quotient $x \longrightarrow -\frac{x^T A x}{x^T x}$. An immediate benefit is that, if $\delta > 0$, the minimizer of (1.2.1.5) is isolated since it is the subspace $\mathcal{V}_\alpha = \operatorname{Span}(V_\alpha)$.

One of the contributions of this thesis is to develop and analyze various solvers for this optimization problem (thus also for the symmetric eigenvalue problem), using off-the-shelf techniques from convex optimization. It turns out that in many cases the outcome is surprisingly competitive in practice, while maintains good theoretical properties. All that is possible via the discovery of a convexity-like structure for (1.2.1.5), analyzed in Section 2.

1.2.2 Polar decomposition

The second important problem presented in this thesis has to do with polar decomposition. The polar decomposition of a matrix is a standard factorization, where some matrix $C \in \mathbb{R}^{n'\times n}$, $n' \geq n$, must be written as the product of an

orthonormal matrix $X \in \mathbb{R}^{n' \times n}$ and a symmetric and positive semi-definite matrix $P \in \mathbb{R}^{n \times n}$, i.e.

$$C = XP.$$

Such a decomposition always exists and a good way to see that is through the singular value decomposition. If a singular value decomposition of C is

$$C = U\Sigma V^T,$$

then the "polar factor" X of the polar decomposition is given as

$$X = UV^T$$

and the symmetric positive semi-definite part P is given as

$$P = V\Sigma V^T.$$

One can easily see that the polar decomposition of C is unique if and only if C is invertible, i.e. if and only if its singular values are all positive.

The most direct way to compute a polar decomposition is via the SVD. Clearly, this approach is too expensive. The numerical linear algebra community has developed plenty of faster algorithms to tackle this problem. The most basic one is the Newton method ([45], Section 8.3). The Newton method is in general fast in the late stage of convergence, but can be very slow at the beginning if the matrix C is ill-conditioned. Another prominent class of algorithms is the Padé family of iterations ([45], Section 8.5), which suffers more or less by the same issues.

Most of the effort in the last few years has been focused on scaling the basic Newton iteration, in order to obtain variants that do not suffer from slow convergence at the beginning of the iterations. The so-called "optimal" scaling [57] enjoys excellent theoretical behaviour, but the scaling factor depends on the (generally unknown) smallest and largest singular values in each iterate X_t. A more practical version, that however lacks convergence guarantees, can be found in [44]. A middle ground with a sub-optimal computable scaling that still enjoys some convergence guarantees can be found in [28].

The state-of-the-art in this area comes probably from [76]. There, the Halley's method (which is a member of the Padè family of iterations) is scaled in a principled way. The Halley method has cubic asymptotic convergence, but the initial stage can be very slow for ill-conditioned matrices [37]. The scaling of [76] helps to improve its performance in the initial stage of convergence.

An interesting property of the polar factor is that it is the closest orthonormal matrix to the original matrix C (see [45], Theorem 8.4). This makes polar decomposition intimately related to the orthogonal Procrustes problem (see [45], Theorem 8.6). The procrustes problem [98] is important in many areas of applied science [6, 36, 55]. It seeks for an orthogonal matrix $X \in \mathbb{O}(n)$, such

that the quantity $\|AX - B\|_F^2$ for two matrices $A, B \in \mathbb{R}^{m\times n}$ is as small as possible. This problem admits the equivalent formulation

$$\min_{X\in\mathbb{O}(n)} -\operatorname{Tr}(CX),$$

with $C := B^T A$, and its solution is the polar factor of the matrix $C^T \in \mathbb{R}^{n\times n}$. This problem turns out to have a geodesic convexity-like structure in the orthogonal group, which we analyze in Section 7. This structure is similar to the one that is analyzed for the symmetric eigenvalue problem in Section 2.

While we do not develop some application for this theory, as we do for the case of the symmetric eigenvalue problem, we predict that many applications can be found in noisy orthogonal Procrustes settings. In general, polar factors behave quite badly with respect to perturbations of the original matrix. Let $\tilde{C}$ be a perturbation of C, then the distance between the polar factors $\tilde{X}$ and X can be upper bounded in general as (see Theorem 8.10 in [45])

$$\|X - \tilde{X}\|_F \leq \frac{2}{\sigma_{\min}(C) + \sigma_{\min}(\tilde{C})}\|C - \tilde{C}\|_F.$$

This means that computing the polar factor of a perturbed version of C fast and in high accuracy does not mean much, especially in the case where C and its perturbed version are nearly singular. In other words, we cannot just take $\tilde{C}$ and apply some of the classic algorithms mentioned above directly on it.

1.3 Basics from differential geometry

In this section, we briefly present basic notions from the field of differential geometry that will be useful later. The main point here is to present a shallow introduction for the readers who are unfamiliar with the basics of differential geometry, focusing on the intuitive relationship with more familiar notions from the geometry of the Euclidean space. For an in depth study of differential geometry, the reader can use a variety of excellent textbooks including [67, 95, 106]. Here, we just follow the simplistic exposition of [10].

We also analyze the geometry of specific manifolds that are useful for our purposes, namely the sphere, the Grassmann manifold and the orthogonal group, this time in mode detail. The classic sources for whatever concerns algorithmic computation on matrix manifolds are [3] and the seminal paper [35].

Manifolds. A differentiable manifold $\mathcal{M}$ is a topological space that is locally Euclidean. This means that for any point $x \in \mathcal{M}$, we can find a neighborhood that is diffeomorphic to an open subset of some Euclidean space. This Euclidean space can be proved to have the same dimension, regardless of the chosen point, called the dimension of the manifold. Considering curves $c : [0, 1] \to \mathcal{M}$ that

pass from a specific point $x \in \mathcal{M}$, the space of their derivatives at x is called the tangent space and is usually denoted by $T_x\mathcal{M}$.

A Riemannian manifold $(\mathcal{M}, g)$ is a differentiable manifold equipped with a Riemannian metric g_x, i.e. an inner product for each tangent space $T_x\mathcal{M}$. We denote the inner product of $u, v \in T_x\mathcal{M}$ with $\langle u, v \rangle_x$ or just $\langle u, v \rangle$ when the tangent space is obvious from context. Similarly, we consider the norm as the one induced by the inner product at each tangent space.

Geodesics Geodesics are curves $\gamma : [0, 1] \to \mathcal{M}$ of constant speed and of (locally) minimum length. They can be thought of as the Riemannian generalization of straight lines in Euclidean spaces. Geodesics are used to construct the exponential map $\mathrm{Exp}_x : T_x\mathcal{M} \to \mathcal{M}$, defined by $\mathrm{Exp}_x(v) = \gamma(1)$, where γ is the unique geodesic such that $\gamma(0) = x$ and $\dot{\gamma}(0) = v$. The exponential map is locally a diffeomorphism. Using the notion of geodesics, we can define an intrinsic distance (denoted as dist) between two points in the Riemannian manifold $\mathcal{M}$, as the infimum of lengths of geodesics that connect these two points. A Riemannian manifold of which any two points are connected by some geodesic is called *complete*. Geodesics also provide a way to transport vectors from one tangent space to another. This operation, called parallel transport, is usually denoted by $\Gamma_x^y : T_x\mathcal{M} \to T_y\mathcal{M}$. Closely linked to geodesics is the notion of injectivity radius. Given a point $x \in \mathcal{M}$, we define the injectivity radius at x (denoted $\mathrm{inj}(x)$) to be the radius of the biggest ball around x that the exponential map Exp_x restricted to it is a diffeomorphism. We denote the inverse of the exponential map inside this ball by Log_x and we call it Riemannian logarithm. Notice that in the Euclidean space the logarithm between two points is just their difference: $\mathrm{Log}_x(y) = y - x$. In general, we have that, if $\mathrm{Log}_x(y)$ is well-defined, then $\mathrm{Log}_x(y) = \mathrm{dist}(x, y)$.

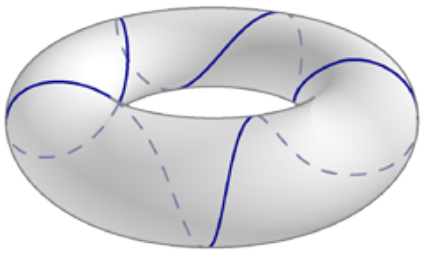


Figure 1.4: The geodesics of the torus, by Mark Irons.

Vector fields and the Riemannian gradient The notion of a vector field is central in calculus. It is also important in Riemannian geometry:

Definition 1.11 *Let $\mathcal{M}$ be a Riemannian manifold. A vector field X in $\mathcal{M}$ is a smooth map $X : \mathcal{M} \to \mathcal{T}\mathcal{M}$, where $\mathcal{T}\mathcal{M}$ is the tangent bundle, i.e. the collection of all tangent vectors in all tangent spaces of $\mathcal{M}$ such that $p \circ X$ is the identity (p is the projection from $\mathcal{T}\mathcal{M}$ to $\mathcal{M}$).*

One can see a vector field as an infinite collection of imaginary curves, the so-called integral curves (formally they are solutions of some first-order differential equations on $\mathcal{M}$).

A prominent vector field for us will be the Riemannian gradient of a real-valued function $f : \mathcal{M} \to \mathbb{R}$:

Definition 1.12 *The Riemannian gradient* $\mathrm{gradf}(x)$ *of a function $f : \mathcal{M} \to \mathbb{R}$ at a point $x \in \mathcal{M}$, is the tangent vector at x, such that $\langle \mathrm{gradf}(x), u \rangle = df(x)u$* [1]*, for any $u \in T_x\mathcal{M}$.*

Covariant differentiation and the Riemannian Hessian The most suitable notion to capture second order changes on a Riemannian manifold is called covariant differentiation and it is induced by the fundamental property of Riemannian manifolds to be equipped with a connection. The fact that a connection can always be defined in a Riemannian manifold is the subject of the so-called fundamental theorem of Riemannian geometry. We are interested in a specific type of connection, called the Levi-Civita connection, which induces a specific type of covariant derivative. For our purpose, it will however be sufficient to define the notion of covariant derivative using the (simpler) notion of parallel transport.

Definition 1.13 *Given two vector fields X, Y in a Riemannian manifold $\mathcal{M}$, we define the covariant derivative of Y along X as*

$$\nabla_X Y(x) := \lim_{t \to 0} \frac{\Gamma_{\gamma(t)}^{\gamma(0)} Y(\gamma(t)) - Y(x)}{h},$$

with γ the unique integral curve of X passing from x.

Given the notions of Riemannian gradient and covariant differentiation, we can define the notion of Riemannian Hessian:

Definition 1.14 *Given vector fields X, Y in $\mathcal{M}$, we define the Hessian operator of f to be*

$$\mathrm{Hess}\, f(X, Y) := \langle \nabla_X \,\mathrm{grad}\, f, Y \rangle.$$

[1] df denotes the differential of f, i.e. $df(x)[u] = \lim_{t \to 0} \frac{f(c(t)) - f(x)}{t}$, where $c : [0, 1] \to \mathcal{M}$ is a smooth curve such that $c(0) = x$ and $\dot{c}(0) = u$.

This $(0,2)$-tensor defines a bilinear form at each tangent space, i.e. $\operatorname{Hess} f(x)$ is a map from $T_x\mathcal{M}$ to $T_x\mathcal{M}$. A simpler definition of the operator $\operatorname{Hess} f(x)$ at some point $x \in \mathcal{M}$ can be

$$\operatorname{Hess} f(x)v := \lim_{t\to 0} \frac{\Gamma^x_{c(t)}\operatorname{grad} f(c(t)) - \operatorname{grad} f(x)}{t},$$

for some curve c, such that $c(0) = x$ and $\dot{c}(0) = v$.

Curvature. The sectional curvatures is a way of measuring the curvature of a Riemannian manifold along a particular 2-dimensional plane within the tangent space at a point.

The sectional curvature K at a point x of a Riemannian manifold $\mathcal{M}$ with Riemannian metric g is defined for each 2-dimensional plane $\sigma \subset T_x\mathcal{M}$. One starts by defining the Riemann curvature tensor $\mathcal{R}$, which is a $(1,3)$-tensor defined as:

$$\mathcal{R}(v,w)z = \nabla_v\nabla_w z - \nabla_w\nabla_v z - \nabla_{[v,w]}z,$$

where ∇ is the Levi-Civita connection, and $v, w, z \in T_x\mathcal{M}$.

The sectional curvature $K(\sigma)$ for the plane σ spanned by v and w is given by:

$$K(\sigma) = K(v,w) = \frac{g(\mathcal{R}(v,w)w, v)}{g(v,v)g(w,w) - g(v,w)^2}.$$

What is important for our purposes is not so much a rigorous definition of sectional curvatures, but rather its implications. All the manifolds that we deal with in this thesis have nonnegative sectional curvatures at all points. This implies the following important geometric bound, which can be seen as a law of cosines for spaces of nonnegative sectional curvatures:

Proposition 1.15 *Consider three points $x, y, z \in \mathcal{M}$ on a manifold of nonnegative sectional curvatures $\mathcal{M}$, such that they are connected by unique geodesics. Then, we have*

1. $\operatorname{dist}^2(x,y) \leq \operatorname{dist}^2(z,x) + \operatorname{dist}^2(z,y) - 2\langle \operatorname{Log}_z(x), \operatorname{Log}_z(y)\rangle$.
2. $\operatorname{dist}(x,y) \leq \| \operatorname{Log}_z(x) - \operatorname{Log}_z(y)\|$.

Proof Both 1 and 2 are simple consequences of the famous Toponogov's theorem (see Theorem 2.2 in [29]). ■

We will need more geometric bounds along the text, which we will present accordingly. We presented the previous bound already, as it will be used extensively.

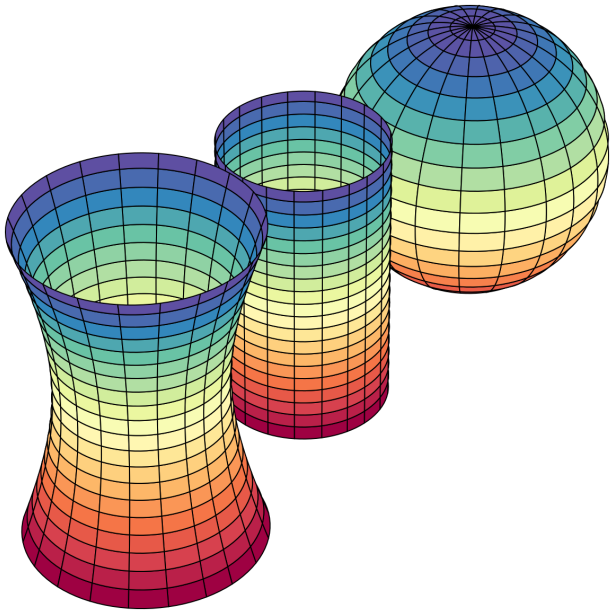


Figure 1.5: Three manifolds of negative, 0 and positive curvature respectively from left to right (taken from Wikipedia).

Geodesic convexity. The differentiability provided by the very structure of a Riemannian manifold is a great feature for generalizing convexity-type notions. It is also suitable for optimizing functions defined over manifolds using gradient-based algorithms. A classic textbook on the topic is [111]. A newer textbook with excellent exposition is [21]. The reader is suggested to consult them in case they need a more complete picture on the relevant notions.

Definition 1.16 *A subset $E \subseteq \mathcal{M}$ of a Riemannian manifold $\mathcal{M}$ is called geodesically uniquely convex, if every two points in E are connected by a unique geodesic.*

Definition 1.17 *A differentiable function $f : \mathcal{M} \to \mathbb{R}$ is called geodesically convex in a geodesically uniquely convex subset E of $\mathcal{M}$, if for all $x, y \in E$ it holds*

$$f(y) - f(x) \geq \langle \text{gradf}(x), \text{Log}_x(y) \rangle.$$

Note that we do not need a function to be differentiable to define convexity. However, since in this thesis all functions of interest will be differentiable, we define convexity directly through the gradient. As in the Euclidean case, any local minimum of a geodesically convex function is a global minimum.
In a similar manner, we define geodesic strong convexity:

Definition 1.18 *A differentiable function $f : \mathcal{M} \to \mathbb{R}$ is called geodesically μ-strongly convex ($\mu > 0$) in a geodesically uniquely convex subset E of $\mathcal{M}$, if for all $x, y \in E$, it holds*

$$f(x) - f(y) \leq \langle \mathrm{gradf}(x), \mathrm{Log}_x(y) \rangle - \frac{\mu}{2}\mathrm{dist}^2(x, y).$$

If a function f is geodesically strongly convex and a minimum exists, then there is only one minimum and it is global.

Definition 1.19 *A function $f : \mathcal{M} \to \mathbb{R}$ defined in a complete manifold $\mathcal{M}$ is called geodesically L-smooth, if for all $x, y \in \mathcal{M}$, it holds*

$$\|\mathrm{gradf}(x) - \Gamma_y^x \mathrm{gradf}(y)\| \leq L\mathrm{dist}(x, y),$$

where Γ_y^x is the parallel transport along some geodesic connecting x and y.

The previous definitions are well constructed enough to imply the standard connection of convexity and smoothness with the Riemannian Hessian:

Proposition 1.20 *A function $f : \mathcal{M} \to \mathbb{R}$ is*

- *geodesically μ-strongly convex in an* open subset *E if and only if*

  $$\mathrm{Hess}f(x) \succeq \mu I$$

  for all $x \in E$.

- *geodesically L-smooth if and only if*

  $$-LI \preceq \mathrm{Hess}f(x) \preceq LI$$

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

$\succeq$ and $\preceq$ represent the classic positive semi-definite order in the space of symmetric matrices.

Geodesic L-smoothness has similar implications with Euclidean L-smoothness.

Proposition 1.21 *If f is L-smooth, then*

- *For all $x, y \in \mathcal{M}$, it holds*

  $$f(y) - f(x) \leq \langle \mathrm{grad}f(x), -\mathrm{Log}_x(y) \rangle + \frac{L}{2}\mathrm{dist}^2(x, y).$$

- *For all $x \in \mathcal{M}$, it holds*

  $$f(x) - f^* \geq \frac{1}{2L}\|\mathrm{grad}f(x)\|^2,$$

  where f^* is the minimum of f.

As discussed previously, in this thesis we are more interested in weaker notions, namely the geodesic Polyak-Łojasiewicz condition and the geodesic weak-quasi-strong convexity. To formally define the above, we just need to substitute the "Euclidean" quantities appearing in their previous definitions with Riemannian analogues of them:

Definition 1.22 *A function $f : \mathcal{M} \to \mathbb{R}$ is:*

- *geodesically Polyak-Łojasiewicz (PL) in a geodesically uniquely convex subset $E \subseteq \mathcal{M}$ if*
$$\|\text{grad} f(x)\|^2 \geq 2\mu(f(x) - f^*),$$
for some $\mu > 0$ and for all $x \in E$.

- *geodesically weak-quasi-strongly convex (WQSC) in a geodesically uniquely convex $E \subseteq \mathcal{M}$, if it has a unique optimum x^* in E and*
$$f(x) - f^* \leq \frac{1}{a}\langle \text{gradf}(x), \text{Log}_x(x^*)\rangle - \frac{\mu}{2}\text{dist}^2(x, x^*),$$
for some $a, \mu > 0$ and for all $x \in E$. Again, if a function is geodesically WQSC with parameters a and μ, we will often write it as geodesically (a, μ)-WQSC. As in the Euclidean case, WQSC implies a PL condition.

Remark 1.3 *We use the term "geodesically" to distinguish between the Euclidean and Riemannian convexity-type notions. However, when the situation is clear from context (i.e. it is obvious we work on a manifold), this word will be omitted.*

The previous notions are suitably constructed in an intrinsic differential-geometric way, such that they give convergence guarantees for a similarly suitable adaptation of gradient descent (1.1.0.2) for a Riemannian manifold:

$$x_{t+1} = \text{Exp}_{x_t}(-\eta \text{grad} f(x_t)),\ x_0 \in \mathcal{M}. \tag{1.3.0.1}$$

This algorithm and variants of it will concern us a lot for the rest of this thesis. We do not give here convergence guarantees of Riemannian gradient descent (1.3.0.1) under the function classes discussed previously, as convergence guarantees will be given throughout the text for specific optimization problems.

Besides gradient descent, Riemannian adaptations of accelerated gradient descent with Nesterov momentum also exist ([4, 58, 75, 119]). In Section 6 we develop a version of such algorithm for the symmetric eigenvalue problem with rigorous convergence guarantees.

1.3.1 The geometry of specific manifolds of interest

1.3.1.1 Sphere

The first manifold that we discuss is the sphere, i.e. the set of vectors of unit norm:

$$\mathbb{S}^{n-1} = \{x \in \mathbb{R}^n / \|x\| = 1\}.$$

This space is useful when one is interested in computing only one eigenvalue and associated eigenvector of a symmetric matrix. We present here some basic quantities regarding the geometry of the sphere and refer the reader to [3, pages 73–76] for a more comprehensive presentation.

Tangent Space: The tangent space of the $(n-1)$-dimensional sphere $\mathbb{S}^{n-1}$ at a point x is an $(n-1)$-dimensional vector space, which generalizes the notion of a two-dimensional tangent plane. We denote it by $T_x\mathbb{S}^{n-1}$ and a vector v belongs in it, if and only if, it can be written as $\dot{c}(0)$, where $c\colon (-\varepsilon, \varepsilon) \to \mathbb{S}^{n-1}$ (for some $\varepsilon > 0$) is a smooth curve with $c(0) = x$. The tangent space at x can be given also in an explicit way, as the set of all vectors in $\mathbb{R}^n$ orthogonal to x with respect to the usual inner product. Given a vector $w \in \mathbb{R}^n$, we can always project it orthogonally in any tangent space of $\mathbb{S}^{n-1}$. Taking all vectors to be column vectors, the orthogonal projection in $T_x\mathbb{S}^{n-1}$ satisfies

$$\mathrm{Proj}_x(w) = (I - xx^T)w.$$

Geodesics: Geodesics on high-dimensional surfaces are defined to be locally length-minimizing curves. On the $(n-1)$-dimensional sphere, they coincide with great circles. These can be computed explicitly and give rise to the exponential and logarithmic maps. These are given by the following well-known formulas

$$\mathrm{Exp}_x(v) = \cos(\|v\|)x + \sin(\|v\|)\frac{v}{\|v\|},\ \mathrm{Log}_x(y) = \arccos(\langle x, y\rangle)\frac{\mathrm{Proj}_x(y-x)}{\|\mathrm{Proj}_x(y-x)\|}. \tag{1.3.1.1}$$

The distance between points x and y measured intrinsically in the sphere is

$$\mathrm{dist}(x, y) = \|\mathrm{Log}_x(y)\| = \arccos(\langle x, y\rangle). \tag{1.3.1.2}$$

Notice that $\langle x, y\rangle = \|x\|\|y\|\cos(\angle(x, y)) = \cos(\angle(x, y))$, thus the distance of x and y is actually the angle between them.

The inner product inherited by the ambient Euclidean space $\mathbb{R}^n$ provides a way of parallel transport. If $y = \mathrm{Exp}_x(tv)$, then parallel transport is given by the formula

$$\Gamma_x^y u = \left(I + \cos(t\|v\| - 1)\frac{vv^T}{\|v\|^2} - \sin(t\|v\|)\frac{pv^t}{\|v\|}\right)u.$$

Riemannian Gradient: The Riemannian gradient (which has been defined previously in general) takes a particularly simple form in the case of the sphere. We can compute the Riemannian gradient by orthogonally projecting the Euclidean gradient $\nabla f(x)$ computed in the ambient space $\mathbb{R}^n$ into the tangent space of x:

$$\mathrm{grad} f(x) = \mathrm{Proj}_x(\nabla f(x)) = (I - xx^T)\nabla f(x).$$

Curvature: The sphere is a manifold of constant sectional curvature, equal to 1. For our purposes, we only use that its sectional curvatures are nonnegative.

1.3.1.2 Grassmann manifold

The (n, k)-Grassmann manifold is defined as the set of all k-dimensional subspaces of $\mathbb{R}^n$:

$$\mathrm{Gr}(n, k) = \{\mathcal{X} \subseteq \mathbb{R}^n : \mathcal{X} \text{ is a subspace and } \dim(\mathcal{X}) = k\}.$$

Any element $\mathcal{X}$ of $\mathrm{Gr}(n, k)$ can be represented by a matrix $X \in \mathbb{R}^{n\times k}$ that satisfies $\mathcal{X} = \mathrm{Span}(X)$. Such a representative is not unique since $Y = XQ$ for some invertible matrix $Q \in \mathbb{R}^{k\times k}$ satisfies $\mathrm{Span}(Y) = \mathrm{Span}(X)$. Without loss of generality, we will therefore always take matrix representatives X of subspaces $\mathcal{X}$ that have orthonormal columns. With some care, the non-uniqueness of the representatives is not a problem [2]. For example, the cost function (1.2.1.5) is invariant to Q.

Tangent space and Riemannian metric: The set $\mathrm{Gr}(n, k)$ admits the structure of a differentiable manifold with tangent spaces

$$T_{\mathcal{X}} \mathrm{Gr}(n, k) = \{G \in \mathbb{R}^{n\times k} : X^T G = 0\}, \tag{1.3.1.3}$$

where $\mathcal{X} = \mathrm{Span}(X)$. Since $X^T G = 0$ if and only if $(XQ)^T G = 0$, for any invertible matrix $Q \in \mathbb{R}^{k\times k}$, this description of the tangent space does not depend on the representative X. However, a specific tangent vector G will depend on the chosen X. With slight abuse of notation [3], the above definition should therefore be interpreted as: given a fixed X, we define tangent vectors $G_1, G_2, \ldots$ of $\mathrm{Gr}(n, k)$ at $\mathcal{X} = \mathrm{Span}(X)$.

This subtlety is important, for example, when defining an inner product on $T_{\mathcal{X}} \mathrm{Gr}(n, k)$:

$$\langle G_1, G_2\rangle_{\mathcal{X}} = \mathrm{Tr}(G_1^T G_2) \quad \text{with } G_1, G_2 \in T_{\mathcal{X}} \mathrm{Gr}(n, k).$$

[2] This can be made very precise by describing $\mathrm{Gr}(n, k)$ as the quotient of the Stiefel manifold with the orthogonal group. The elegant theory of this quotient manifold is worked out in [3].

[3] Using the quotient manifold theory, one would use horizontal lifts.

Here, G_1 and G_2 are tangent vectors of the same representative X. Observe that the inner product is invariant to the choice of orthonormal representative: If $\bar{G}_1 = G_1 Q$ and $\bar{G}_2 = G_2 Q$ with orthogonal Q, then we have

$$\langle \bar{G}_1, \bar{G}_2 \rangle_{\mathcal{X}} = \operatorname{Tr}(\bar{G}_1^T \bar{G}_2) = \operatorname{Tr}(Q^T G_1^T G_2 Q) = \operatorname{Tr}(G_1^T G_2 Q Q^T) = \operatorname{Tr}(G_1^T G_2).$$

It is easy to see that the norm induced by this inner product in any tangent space is the Frobenius norm, which we will denote as $\|\cdot\| := \|\cdot\|_F$.

The orthogonal projection of a matrix $W \in \mathbb{R}^{n \times k}$ onto the tangent space $T_{\mathcal{X}} \operatorname{Gr}(n,k)$ is

$$\operatorname{Proj}_{\mathcal{X}}(W) = (I - XX^T)W,$$

where X is an orthonormal representative of $\mathcal{X}$.

Exponential map and Riemannian logarithm: Given the Riemannian structure of $\operatorname{Gr}(n,k)$, we can compute the exponential map at a point $\mathcal{X}$ as [2, Thm. 3.6]

$$\begin{aligned} \operatorname{Exp}_{\mathcal{X}} : T_{\mathcal{X}} \operatorname{Gr}(n,k) &\to \operatorname{Gr}(n,k) \\ G &\mapsto \operatorname{Span}(\, XV \cos(\Sigma) + U \sin(\Sigma)\,), \end{aligned} \tag{1.3.1.4}$$

where $U\Sigma V^T$ is the *compact* SVD of G such that Σ and V are square matrices.

The exponential map is invertible in the domain [18, Prop. 5.1]

$$\left\{ G \in T_{\mathcal{X}} \operatorname{Gr}(n,k) \colon \|G\|_2 < \frac{\pi}{2} \right\}, \tag{1.3.1.5}$$

where $\|G\|_2$ is the spectral norm of G. The inverse of the exponential map restricted to this domain is the logarithmic map, denoted by Log. Given two subspaces $\mathcal{X}, \mathcal{Y} \in \operatorname{Gr}(n,k)$, we have

$$\operatorname{Log}_{\mathcal{X}}(\mathcal{Y}) = U \operatorname{atan}(\widehat{\Sigma})\, V^T, \tag{1.3.1.6}$$

where $U\widehat{\Sigma}V^T = (I - XX^T)Y(X^TY)^{-1}$ is again a compact SVD. This is well-defined if X^TY is invertible, which is guaranteed if all principal angles between $\mathcal{X}$ and $\mathcal{Y}$ are strictly less than $\pi/2$. By taking $G = \operatorname{Log}_{\mathcal{X}}(\mathcal{Y})$, we see that $\Sigma = \operatorname{atan}(\widehat{\Sigma})$. We can express the Riemannian logarithm using the notion of principal angles between subspaces. The intrinsic distance induced by the aforementioned Riemannian metric is

$$\operatorname{dist}(\mathcal{X}, \mathcal{Y}) = \|\operatorname{Log}_{\mathcal{X}}(\mathcal{Y})\| = \|\operatorname{Log}_{\mathcal{Y}}(\mathcal{X})\| = \sqrt{\theta_1^2 + \ldots + \theta_k^2} = \|\theta\|_2, \tag{1.3.1.7}$$

where $\theta = (\theta_1, \ldots, \theta_k)^T$ with θ_j being the principal angles between the subspaces $\mathcal{X}$ and $\mathcal{Y}$. For more details on these facts, the reader can refer to Section 4.3 in [35] (arc length distance).

Riemannian gradient: The gradient of a function $f : \mathrm{Gr}(n,k) \to \mathbb{R}$ at a point $\mathcal{X}$ is given as the orthogonal projection of the Euclidean gradient $\nabla f(X)$ computed in the ambient space at an orthonormal representative X of $\mathcal{X}$:

$$\mathrm{grad}\, f(\mathcal{X}) = (I - XX^T)\nabla f(X).$$

Curvature: We can compute exactly the sectional curvatures in $\mathrm{Gr}(n,k)$, but for our purposes we only need that they are everywhere non-negative [18, 113]. This means that the geodesics on the Grassmann manifold spread more slowly than in Euclidean space, which is essentially quantified by Proposition 1.15.

1.3.1.3 The orthogonal group

The orthogonal group $\mathbb{O}(n)$ is the set of all orthogonal matrices in $\mathbb{R}^{n\times n}$. It is a Riemannian manifold and a group, i.e. it has the structure of a Lie group. The orthogonal group is disconnected, with two connected components, namely, the orthogonal matrices with determinant equal to 1 and the ones with determinant equal to -1. We present again the basics of the geometry of this manifold and refer the reader to [18] for more.

Tangent space and Riemannian metric: The tangent space at a point $X \in \mathbb{O}(n)$ is

$$T_X\mathbb{O}(n) = \{X\Omega \,/\, \Omega \in \mathbb{R}^{n\times n} \text{ is skew-symmetric, i.e. } \Omega^T = -\Omega\}.$$

The most usual Riemannian metric that one equipes this space is

$$\langle V, W\rangle_X := \mathrm{Tr}(W^T V).$$

Given this Riemannian metric, the orthogonal projection of a matrix $Z \in \mathbb{R}^{n\times n}$ onto $T_X\mathbb{O}(n)$ is

$$\mathrm{Proj}_X(Z) = X\mathrm{skew}(X^T Z),$$

where

$$\mathrm{skew}(A) := \frac{A - A^T}{2}$$

is the skew-symmetric part of a matrix.

Exponential map and Riemannian logarithm: The exponential map at a point X in the direction $X\Omega$ is defined as

$$\mathrm{Exp}_X(X\Omega) = X\exp_m(\Omega),$$

where $\exp_m$ is the matrix exponential.
The Riemannian logarithm is the inverse of the exponential map, when the latter is invertible. We now examine when this is the case.

In order to identify the domain where the exponential map is invertible, we need to verify when the equation

$$\mathrm{Exp}_X(X\Omega) = X \exp_m(\Omega) = Y$$

has a unique solution. This happens if and only if the equation

$$\exp_m(\Omega) = X^T Y$$

has a unique solution. Consider the eigenvalue decomposition $\Omega = U\Lambda U^{-1}$, where Λ is diagonal with entries of the form $i\theta$ (since Ω is skew-symmetric). This implies that the eigenvalue decomposition of $\exp_m(\Omega)$ is $U \exp_m(\Lambda) U^{-1}$ and $\exp_m(\Lambda)$ is diagonal featuring entries of the form $e^{i\theta}$ with $\theta \in (-\pi, \pi]$. Thus, the previous equation boils down to a series of equations of the form

$$e^{i\theta} = s,$$

where s are the eigenvalues of $X^T Y$. These equations are well-defined and have a unique solution if and only if s is in the domain of a definition of the complex logarithm, i.e. in $\mathbb{C} \setminus (-\infty, 0]$. In that case, θ is allowed to be in $(-\pi, \pi)$, i.e $\theta \neq \pi$. We can summarize the previous discussion as follows:

Lemma 1.23 • *The domain of the orthogonal group where the exponential map is a diffeomorphishm is*

$$\{X\Omega / \Omega^T = -\Omega, \|\Omega\|_2 < \pi\}. \tag{1.3.1.8}$$

- *Let $X, Y \in \mathbb{O}(n)$. If the phases θ of the eigenvalues $e^{i\theta}$ of $X^T Y$ satisfy $\theta \in (-\pi, \pi)$, then there is a unique geodesic connecting X and Y. In this case, it trivially holds that X and Y are in the same connected component of $\mathbb{O}(n)$.*
- *If some of the θ's are equal to π, then it holds: if there is even number of θ's equal to π, then X and Y are in the same connected component (and are connected by multiple geodesics). If there is odd number of θ's equal to π, then X and Y are in different connected components (i.e.* $\det(XY) = -1$*).*

Let us now consider X and Y such that $X^T Y$ has eigenvalues with phases in $(-\pi, \pi)$. Then $\mathrm{Log}_X(Y)$ is well-defined and

$$\mathrm{Exp}_X(\mathrm{Log}_X(Y)) = Y.$$

We can write $\mathrm{Log}_X(Y) = X\Omega$ for some skew-symmetric Ω and we have

$$X \exp_m(\Omega) = Y, \tag{1.3.1.9}$$

which can be written as

$$\Omega = \log_m(X^T Y),$$

where $\log_m$ is the matrix logarithm.

Thus,

$$\mathrm{Log}_X(Y) = X \log_m(X^T Y). \tag{1.3.1.10}$$

Note that $\log_m(X^T Y)$ is indeed a skew-symmetric matrix since X and Y are orthogonal.

Parallel transport: In the orthogonal group, the parallel transport from a point X to a point Y (denoted by Γ_X^Y), is given by

$$\Gamma_X^Y(X\Omega) = Y(X^T Y \Omega Y^T X).$$

Notice that $X^T Y \Omega Y^T X$ is a skew-symmetric matrix, since it is a conjugation of the skew-symmetric matrix Ω. This definition makes sense of course only if X and Y are in the same connected component of $\mathbb{O}(n)$.

Riemannian distance: Since we have computed the Riemannian logarithm between two orthogonal matrices X and Y, we can also compute the Riemannian distance between such matrices based on it:

$$\mathrm{dist}^2(X,Y) = \| \mathrm{Log}_X(Y)\|^2 = \|X \log_m(X^T Y)\|^2 = \| \log_m(X^T Y)\|^2.$$

In order to proceed, we decompose the orthogonal matrix $X^T Y$ into the so-called canonical form PDP^T, where P is an orthogonal matrix featuring the eigenvectors of $X^T Y$ in its columns and D is block diagonal. D is constructed as follows. When $X^T Y$ has an eigenvalue equal to 1, D has a diagonal entry equal to 1. When $X^T Y$ has an eigenvalue of the form $e^{i\theta}$ for some $\theta \in (-\pi, 0) \cup (\pi, 0)$, then $e^{-i\theta}$ is also an eigenvalue and D features the 2×2 block that is the 2-d rotation with angle θ. That is $\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$.

The matrix logarithm has the following convenient property. Given the above decomposition, we have

$$\log_m(PDP^T) = P \log_m(D) P^T.$$

Taking D as constructed previously, $\log_m(D)$ has 0 in the positions where D has 1 and $\begin{bmatrix} 0 & -\theta \\ \theta & 0 \end{bmatrix}$ where D has $\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$. Since P is orthogonal, the distance between X and Y turns out to be equal to $\| \log_m(D)\|^2 = \mathrm{Tr}(\log_m(D)^T \log_m(D))$. $\log_m(D)^T \log_m(D)$ is again a 2×2 block diagonal matrix with 0's where $\log_m(D)$ has 0's and $\begin{bmatrix} \theta^2 & 0 \\ 0 & \theta^2 \end{bmatrix}$ where $\log_m(D)$ has $\begin{bmatrix} 0 & -\theta \\ \theta & 0 \end{bmatrix}$. Thus, the distance between X and Y is

$$\mathrm{dist}(X,Y) = \left(\sum_{i=1}^{n} \theta_i^2 \right)^{1/2}, \tag{1.3.1.11}$$

where $e^{i\theta_i}$ are the eigenvalues of X^TY. That is to say that

$$\text{dist}(X, Y) = \|\phi\|_2,$$

where $\phi = (\theta_1, \ldots, \theta_n)$. If $\theta_j = 0$, then it appears only once in ϕ, otherwise it appears as a couple with $-\theta_j$. Note that with a simple limit argument, we can conclude that the same formula still holds when some phases of the eigenvalues of X^TY are equal to π.

Riemannian gradient: As in the previous cases of embedded submanifolds, the gradient of a function $f : \mathbb{O}(n) \to \mathbb{R}$ is the orthogonal projection of the Euclidean gradient in the relevant tangent space:

$$\text{grad} f(X) = \text{Proj}_X(\nabla f(X)) = X\text{skew}(X^T \nabla f(X)).$$

Curvature: The sectional curvatures in the orthogonal group are nonnegative, as the orthogonal group is a special case of a Stiefel manifold and all Stiefel manifolds have nonnegative sectional curvatures. This means that Proposition 1.15 holds for the orthogonal group.

1.4 Matching of sections to published or under review work

For making the study of this thesis easier, we present here the matching of each section to work of us that is online. Some of these papers are already published, while other are still under review.

- Section 2 $\longleftrightarrow$ [14] (published)
- Section 3 $\longleftrightarrow$ [8] (published)
- Section 4 $\longleftrightarrow$ [9] (under review)
- Section 5 $\longleftrightarrow$ [12] (published)
- Section 6 $\longleftrightarrow$ [15] (under review)
- Section 7 $\longleftrightarrow$ [13] (under preparation)
- Section 8 $\longleftrightarrow$ [7] (published)

Most of these papers have been written in collaboration with very capable colleagues, who we would like to thank. Important role in their quality has been played also by various reviewers, whose diligence has been of great service for us.

2 Geodesic convexity of the symmetric eigenvalue problem and convergence of gradient descent

We start the main part of this thesis with a thorough study of the optimization landscape of problem (1.2.1.5). This section follows the exposition of our work [14]. We reveal a convexity-like structure that on the one hand explains why eigenvalue problems are easy to solve, while on the other hand is useful for algorithmic computation.

2.1 Introduction

First, we discuss some related works. As discussed in the introduction, the symmetric eigenvalue problem has been popular for several decades in the numerical linear algebra and optimization communities. When only a few eigenvalues are targeted, the main solvers for this problem have been based on subspace iteration and Krylov subspace methods. Less but still considerable attention has been given to the gradient descent method and its accelerated versions. Most works on gradient descent focus only on computing the first leading eigenvector of a symmetric matrix ($k = 1$), using a Euclidean version of the algorithm. Asymptotic convergence rates are known for this setting since the 1950's, see [43]. More recently, exact non-asymptotic estimates for the same Euclidean gradient descent with exact line search were proved in [62]. For a more comprehensive overview of this line of research, the reader can refer to [85] and the references therein.

Regarding the block version of the algorithm, where one targets multiple pairs of eigenvalues and eigenvectors, much less is known. We refer here to [86], which presents a gradient descent-like method for the multiple eigenvector problem using Ritz projections onto a $2k$-dimensional subspace in each step. The convergence of this algorithm is proved to be linear, but computing the Ritz projections is quite expensive. Instead, in this section, we consider a much cheaper version of gradient descent by directly choosing only one of the vectors in this $2k$-dimensional subspace to update our algorithm. Some analysis for such a gradient descent (without Ritz projection) on the Grassmann manifold using a retraction and an Armijo step size is provided in [3] (see Algorithm 3 and Theorem 4.9.1). Unfortunately this convergence rate is asymptotic, that is, a linear rate is achieved after an unknown number of iterations. The region in which the convergence happens cannot be quantified. Also, such a convergence rate does not yield an iteration complexity for the algorithm.

The optimization landscape provided by the block Rayleigh quotient on the Grassmann manifold has also received some attention lately. [97] provides many interesting properties of the critical points of this function and proves that all but the global optimum are strict saddles. This is later used to derive favourable convergence properties for a hybrid method consisting of Riemannian gradient

descent in a first stage and a Riemannian Newton's method in a final stage. [71] proves the so-called robust strict saddle property for this function, that is, the Hessian evaluated in each critical point except the global optimum has both positive and negative eigenvalues in a whole neighborhood. However, none of these papers talk about (generalized) convexity of any form, nor discusses any convergence rates for gradient descent.

Turning the discussion to the convexity properties of eigenvalue problems, there is a new line of research concerned by that. In [117], the authors prove (Theorem 4) that the Rayleigh quotient is geodesically PL in the sphere ($k = 1$), that is, it satisfies a spherical version of the Polyak–Łojasiewicz inequality. The result of [117] is strengthened by our work [8], which is a special case of the work presented in this section and will be discussed in detail (from an application point of view) in the next section. Finally, [5] examines (among other contributions) the convexity structure of the same block version of the symmetric eigenvalue problem on the Grassmann manifold that we introduced above. Unfortunately, the characterization of the geodesic convexity region independently of the spectral gap δ (Corollary 5 in [5]) is wrong (see our Section 2.5 for a counterexample). As we will prove in Theorem 2.21, the geodesic convexity region of f (and the one of the equivalent cost function used in [5]) needs to depend on the spectral gap, as appears also in [50, Lemma 7] in the case of the sphere ($k = 1$).

To the best of our knowledge, the work presented in this section is the first one that provides non-asymptotic convergence rates for the gradient descent algorithm for the multiple eigenvalue-eigenvector problem on the Grassmann manifold. We do so by first proving that problem (1.2.1.5) satisfies a WQSC condition.

As mentioned above, the standard algorithm for computing the leading eigenspace of dimension k is subspace iteration (or power method when $k = 1$).[4] However, there are reasons to believe that, in certain cases, Riemannian gradient descent (and its accelerated version with non-linear conjugate gradients) should be preferred, especially in noisy settings [8] or in electronic structure calculations where the leading eigenspace of many varying matrices A needs to be computed.[5] In particular, [8] presents strong experimental evidence that gradient descent is more robust to perturbations of the matrix-vector products than subspace iteration close to the optimum. While subspace iteration still behaves better at the start of the iteration, it asymptotically fails to converge to an approximation of the leading subspace that is as good as the one estimated by Riemannian gradient descent. While [8] dealt with a noisy situation due to calculations in a distributed setting with limited communication, exactly the same effect can

[4] Krylov methods are arguably the most popular algorithms but they do not iterate on a subspace directly and are typically started from a single vector. In particular, they cannot easily improve a given approximation of a subspace for large $k > 1$.

[5] More on that in Section 5.

be observed when we inject the matrix-vector products with Gaussian noise. Thus, we expect gradient descent to perform better than subspace iteration close to the optimum in any stochastic regime [42].

Regarding worst-case theoretical guarantees, the strongest convergence result for subspace iteration in the presence of a strictly positive spectral gap δ is in terms of the largest principal angle between the iterates and the optimum [39], that is, the ℓ_∞-norm of the vector of principal angles. In contrast, our convergence result for gradient descent for $\delta > 0$ (Theorem 2.10) is in terms of the ℓ_2-norm of the same vector of angles, which is in general stronger. When $\delta = 0$, it is known from [65, 91] that the largest eigenvalue ($k = 1$) can still be efficiently estimated. We extend this result for $k > 1$ and prove a convergence rate for gradient descent for the function values of f (Theorem 2.12), relying only on weak-quasi convexity (and thus using a different argument from [65, 91]). Weak-quasi convexity can be seen as $(a, 0)$-WQSC.

Block Rayleigh quotient. As discussed in the introduction, the symmetric eigenvalue problem can be transformed into an optimization problem of the block version of the Rayleigh quotient:

$$f(\mathcal{X}) = -\operatorname{Tr}(X^T A X) \text{ where } \mathcal{X} = \operatorname{Span}(X) \in \operatorname{Gr}(n, k) \text{ s.t. } X^T X = I_k.$$

This function has $\mathcal{V}_\alpha = \operatorname{Span}(\begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix})$ as global minimizer. This minimizer is unique on $\operatorname{Gr}(n, k)$ if and only if the spectral gap $\delta := \lambda_k - \lambda_{k+1}$ is strictly positive.

For a given representative X of $\mathcal{X}$, the Riemannian gradient of the block Rayleigh quotient satisfies

$$\operatorname{grad} f(\mathcal{X}) = -2(I - XX^T)AX.$$

Using the notions of the Riemannian gradient and Levi-Civita connection, we can define also a Riemannian notion of Hessian as discussed in the introduction. For the block Rayleigh quotient f, the Riemannian Hessian $\operatorname{Hess} f$ evaluated as bilinear form satisfies

$$\operatorname{Hess} f(\mathcal{X})[G, G] = 2\langle G, GX^T AX - AG\rangle, \tag{2.1.0.1}$$

for $G \in T_{\mathcal{X}} \operatorname{Gr}(n, k)$; see [35, §4.4] or [3, §6.4.2].

2.2 Convexity-like properties of the block Rayleigh quotient

We now prove the new analytic properties of the block Rayleigh quotient $f(\mathcal{X}) = -\operatorname{Tr}(X^T A X)$. These are important in their own right but will also be used later for the convergence of the Riemannian gradient descent method.

2.2.1 Smoothness

A C^2 function defined on the Grassmann manifold is L-smooth (see again Definition 1.19) if the eigenvalues of its Riemannian Hessian are everywhere upper bounded in absolute value by a positive constant L. This is true for the block Rayleigh quotient, as we show in the next proposition:

Proposition 2.1 (Smoothness) *The eigenvalues of the Riemannian Hessian of f on $\mathrm{Gr}(n,k)$ are upper bounded in absolute value by $L := 2(\lambda_1 - \lambda_n)$.*

Proof Let G be a tangent vector of $\mathrm{Gr}(n,k)$ at X. Then the Riemannian Hessian satisfies (see (2.1.0.1))

$$\tfrac{1}{2}\operatorname{Hess} f(\mathcal{X})[G,G] = \operatorname{Tr}(G^T G X^T A X) - \operatorname{Tr}(AGG^T).$$

Since A, X^TAX, GG^T, and G^TG are all symmetric and positive semi-definite matrices, standard trace inequality (see, e.g, [47, Thm. 4.3.53]) gives

$$\operatorname{Hess} f(\mathcal{X})[G,G] \leq 2(\lambda_{\max}(X^TAX) - \lambda_{\min}(A))\|G\|^2.$$

Since X has orthonormal columns, $\lambda_{\max}(X^TAX) \leq \lambda_{\max}(A)$; see, e.g., [47, Cor. 4.3.37]. Thus,

$$\operatorname{Hess} f(\mathcal{X})[G,G] \leq 2(\lambda_1 - \lambda_n)\|G\|^2.$$

Similarly,

$$\begin{aligned}
-\tfrac{1}{2}\operatorname{Hess} f(\mathcal{X})[G,G] &= -\operatorname{Tr}(G^TGX^TAX) + \operatorname{Tr}(AGG^T) \\
&\leq (-\lambda_{\min}(X^TAX) + \lambda_{\max}(A))\|G\|^2 \\
&\leq (-\lambda_{\min}(A) + \lambda_{\max}(A))\|G\|^2 \\
&= (\lambda_1 - \lambda_n)\|G\|^2.
\end{aligned}$$

The last inequality follows from the fact that $\lambda_{\min}(X^TAX) \geq \lambda_{\min}(A)$ (see for instance the Cauchy interlacing theorem).

Putting it all together, we have

$$|\operatorname{Hess} f(\mathcal{X})[G,G]| \leq 2(\lambda_1 - \lambda_n)\|G\|^2$$

and the desired result follows.

■

The result in Proposition 2.1 is tight: Choosing $X = V_\alpha$ and $G = v_n e_1^T$, it is readily verified that the upper bound is attained. From now on, we refer to L as the specific value $2(\lambda_1 - \lambda_n)$. This value also features in a useful upper bound for the spectral norm of the gradient. This bound is independent of $\mathcal{X}$:

Lemma 2.2 *For all $\mathcal{X} \in \mathrm{Gr}(n,k)$ and $L = 2(\lambda_1 - \lambda_n)$, the Riemannian gradient of f satisfies*

$$\| \operatorname{grad} f(\mathcal{X})\|_2 \leq \frac{L}{2}.$$

Proof Since X has orthonormal columns, we can complete it to the orthogonal matrix $Q = \begin{bmatrix} X & X_\perp \end{bmatrix}$. Hence, $\| \operatorname{grad} f(\mathcal{X})\|_2 = \|2(I - XX^T)AX\|_2 = 2\|X_\perp^T AX\|_2$. The result now follows directly from [69, Thm. 2] since A is real symmetric and the definition of $L = 2(\lambda_1 - \lambda_n)$. ■

By the second-order Taylor expansion of f (see, e.g., [21], Corollary 10.54) it is easy to see that Proposition 2.1 implies

$$f(\mathcal{X}) \leq f(\mathcal{Y}) + \langle \operatorname{grad} f(\mathcal{Y}), \operatorname{Log}_{\mathcal{Y}}(\mathcal{X})\rangle + \frac{L}{2} \operatorname{dist}^2(\mathcal{X}, \mathcal{Y}), \tag{2.2.1.1}$$

for any $\mathcal{X}, \mathcal{Y} \in \mathrm{Gr}(n,k)$ such that $\operatorname{Log}_{\mathcal{X}}(\mathcal{Y})$ is well-defined.

As in the introduction, denote the global minimum of f by f^* which is attained at $\mathcal{V}_\alpha \in \mathrm{Gr}(n,k)$. Inequality (2.2.1.1) leads to the following lemma:

Lemma 2.3 *For any $\mathcal{X} \in \mathrm{Gr}(n,k)$ and $L = 2(\lambda_1 - \lambda_n)$, we have*

$$f(\mathcal{X}) - f^* \geq \frac{1}{2L} \| \operatorname{grad} f(\mathcal{X})\|^2.$$

Proof Since f^* is a global minimum of f, we have from (2.2.1.1) that

$$f^* \leq f(\mathcal{X}) \leq f(\mathcal{Y}) + \langle \operatorname{grad} f(\mathcal{Y}), \operatorname{Log}_{\mathcal{Y}}(\mathcal{X})\rangle + \frac{L}{2} \| \operatorname{Log}_{\mathcal{Y}}(\mathcal{X})\|^2,$$

for any $\mathcal{X}, \mathcal{Y} \in \mathrm{Gr}(n,k)$ such that $\operatorname{Log}_{\mathcal{X}}(\mathcal{Y})$ is well-defined.

We set $\mathcal{X} := \operatorname{Exp}_{\mathcal{Y}}\left(-\frac{1}{L}\operatorname{grad} f(\mathcal{Y})\right)$. By Lemma 2.2, we have that $\left\|-\frac{1}{L}\operatorname{grad} f(\mathcal{Y})\right\|_2 < \frac{\pi}{2}$ and by equation (1.3.1.5) we have that $\operatorname{Log}_{\mathcal{Y}}(\mathcal{X})$ is well-defined and equal to $-\frac{1}{L}\operatorname{grad} f(\mathcal{Y})$. Then, the right hand side of the initial inequality becomes

$$f^* \leq f(\mathcal{Y}) - \frac{1}{L}\|\operatorname{grad} f(\mathcal{Y})\|^2 + \frac{1}{2L}\|\operatorname{grad} f(\mathcal{Y})\|^2 = f(\mathcal{Y}) - \frac{1}{2L}\|\operatorname{grad} f(\mathcal{Y})\|^2.$$

Rearranging the last inequality and substituting $\mathcal{Y} = \mathcal{X}$, we get the desired result. ■

Note that we have already discussed these results in a more general regime in Proposition 1.21, but without proof. This is the reason that we discuss them here in more detail.

2.2.2 Weak-quasi convexity and quadratic growth

We now turn our interest in the convexity properties of the block Rayleigh quotient function. We start by proving a property which is known in the literature as *quadratic growth*.

Proposition 2.4 (Quadratic growth) *Let $0 \leq \theta_1 \leq \cdots \leq \theta_k < \pi/2$ be the principal angles between the subspaces $\mathcal{X}$ and $\mathcal{V}_\alpha$. The function f satisfies*

$$f(\mathcal{X}) - f^* \geq c_Q \, \delta \, \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha)$$

where $c_Q = 4/\pi^2 > 0.4$.

Proof

The spectral decomposition of $A = V_\alpha \Lambda_\alpha V_\alpha^T + V_\beta \Lambda_\beta V_\beta^T$ implies

$$X^T A X = X^T V_\alpha \Lambda_\alpha V_\alpha^T X + X^T V_\beta \Lambda_\beta V_\beta^T X. \tag{2.2.2.1}$$

Since $f(\mathcal{X}) = -\operatorname{Tr}(X^T A X)$, we have

$$\begin{aligned} f(\mathcal{X}) - f^* &= \operatorname{Tr}(\Lambda_\alpha) - \operatorname{Tr}(X^T V_\alpha \Lambda_\alpha V_\alpha^T X) - \operatorname{Tr}(X^T V_\beta \Lambda_\beta V_\beta^T X) \\ &= \operatorname{Tr}(\Lambda_\alpha) - \operatorname{Tr}(\Lambda_\alpha V_\alpha^T X X^T V_\alpha) - \operatorname{Tr}(\Lambda_\beta V_\beta^T X X^T V_\beta) \\ &= \operatorname{Tr}(\Lambda_\alpha (I_k - V_\alpha^T X X^T V_\alpha)) - \operatorname{Tr}(\Lambda_\beta V_\beta^T X X^T V_\beta). \end{aligned}$$

From Definition 1.9 of the principal angles between X and V_α, we recall that

$$V_\alpha^T X = U_1 \cos\theta \, V_1^T, \tag{2.2.2.2}$$

where $\cos\theta = \operatorname{diag}(\cos\theta_1, \ldots, \cos\theta_k)$ is a diagonal matrix and U_1, V_1 are orthogonal matrices. Plugging this equality in, we get that the jth eigenvalue of the matrix $I_k - V_\alpha^T X X^T V_\alpha$ is equal to $1 - \cos^2\theta_j = \sin^2\theta_j \geq 0$. Thus, by standard trace inequality for symmetric and positive definite matrices (see, e.g., [47, Thm. 4.3.53]), the first summand above satisfies

$$\operatorname{Tr}(\Lambda_\alpha (I_k - V_\alpha^T X X^T V_\alpha)) \geq \lambda_k \sum_{j=1}^{k} \sin^2\theta_j.$$

The matrix $V_\beta^T X X^T V_\beta$ has the same non-zero eigenvalues with the same multiplicity as the matrix

$$X^T V_\beta V_\beta^T X = I_k - V_1 \cos^2\theta \, V_1^T = V_1 \sin^2\theta \, V_1^T$$

where we used $V_\beta V_\beta^T = I_n - V_\alpha V_\alpha^T$ and the SVD of $V_\alpha^T X$. Thus the jth eigenvalue of $V_\beta^T X X^T V_\beta$ is $\sin^2\theta_j \geq 0$. By trace inequality again, the second summand therefore satisfies

$$\operatorname{Tr}(\Lambda_\beta V_\beta^T X X^T V_\beta) \leq \lambda_{k+1} \sum_{j=1}^{k} \sin^2\theta_j.$$

Putting both bounds together, we get

$$f(\mathcal{X}) - f^* \geq (\lambda_k - \lambda_{k+1}) \sum_{j=1}^{k} \sin^2 \theta_j \geq \delta \sum_{j=1}^{k} \frac{4}{\pi^2} \theta_j^2$$

and the proof is complete by the Definition (1.3.1.7) of dist. ■

Recalling Definition 1.17, we say that f is geodesically convex if for all $\mathcal{X}$ and $\mathcal{Y}$ in a suitable region it holds

$$f(\mathcal{X}) - f(\mathcal{Y}) \leq \langle \text{grad} f(\mathcal{X}), -\text{Log}_{\mathcal{X}}(\mathcal{Y}) \rangle.$$

In Section 2.5, we prove that our objective function f is geodesically convex only in a small neighbourhood of size $\mathcal{O}(\sqrt{\delta})$ around the minimizer $\mathcal{V}_\alpha$. Fortunately, our key result of this section shows that f satisfies a much weaker notion of geodesic convexity, known in the literature as *weak-quasi convexity*, that does not depend on the spectral gap δ.

We first need the following lemma which is a general version of the CS decomposition but applied to our setting of square blocks.

Lemma 2.5 *Let $X, Y \in \mathbb{R}^{n \times k}$ be such that $X^T X = Y^T Y = I_k$ with $k < n$. Choose $X_\perp, Y_\perp \in \mathbb{R}^{n \times (n-k)}$ such that $X_\perp^T X_\perp = Y_\perp^T Y_\perp = I_{n-k}$ and $\text{Span}(X_\perp) = \text{Span}(X)^\perp$, $\text{Span}(Y_\perp) = \text{Span}(Y)^\perp$. Then there exist $0 \leq r, s \leq k$ such that*

$$Y^T X = U_1 \begin{bmatrix} I_r & & \\ & C_s & \\ & & O_{p \times p} \end{bmatrix} V_1^T, \quad Y^T X_\perp = U_1 \begin{bmatrix} O_{r \times m} & & \\ & S_s & \\ & & I_p \end{bmatrix} V_2^T$$

$$Y_\perp^T X = U_2 \begin{bmatrix} O_{m \times r} & & \\ & S_s & \\ & & I_p \end{bmatrix} V_1^T, \quad Y_\perp^T X_\perp = U_2 \begin{bmatrix} -I_m & & \\ & -C_s & \\ & & O_{p \times p} \end{bmatrix} V_2^T$$

with $p = k - r - s$ and $m = n - 2k + r$, and we have

- *orthogonal matrices U_1, V_1 of size k and U_2, V_2 of size $n - k$;*
- *identity matrices I_q of size q;*
- *zero matrices $O_{q \times t}$ of size $q \times t$;*
- *diagonal matrices $C_s = \text{diag}(\alpha_1, \ldots, \alpha_s)$ and $S_s = \text{diag}(\beta_1, \ldots, \beta_s)$ such that $1 > \alpha_1 \geq \cdots \geq \alpha_s > 0$, $0 < \beta_1 \leq \cdots \leq \beta_s < 1$ and $C_s^2 + S_s^2 = I_s$.*

Proof Since $\begin{bmatrix} X & X_\perp \end{bmatrix}$ and $\begin{bmatrix} Y & Y_\perp \end{bmatrix}$ are orthogonal, the result follows directly from the CS decomposition of the orthogonal matrix $P = \begin{bmatrix} Y & Y_\perp \end{bmatrix}^T \begin{bmatrix} X & X_\perp \end{bmatrix}$; see the Theorem of §4 in [92]. ■

Observe that the matrix $\mathrm{diag}(I_r, C_s, O_{p\times p})$ in this lemma corresponds to the matrix $\cos(\theta)$ in Definition 1.9 with θ the vector of principal angles $0 \leq \theta_1 \leq \cdots \leq \theta_k \leq \pi/2$ between $\mathrm{Span}(X)$ and $\mathrm{Span}(Y)$. However, the lemma explicitly splits off the angles that are zero and $\pi/2$ so that it can formulate the related decompositions for $Y^T X_\perp, Y_\perp^T X$, and $Y_\perp^T X_\perp$ with C_s and S_s.

We are now ready to state our weak-quasi convexity result. In the statement of the proposition below (and throughout this section), we use the convention that $\frac{0}{\tan 0} = 1$.

Proposition 2.6 (Weak-quasi convexity) *Let $0 \leq \theta_1 \leq \cdots \leq \theta_k < \pi/2$ be the principal angles between the subspaces $\mathcal{X}$ and $\mathcal{V}_\alpha$. Then, f satisfies*

$$2a(\mathcal{X})\,(f(\mathcal{X}) - f^*) \leq \langle \mathrm{grad} f(\mathcal{X}), -\mathrm{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)\rangle$$

with $a(\mathcal{X}) := \theta_k / \tan\theta_k$.

Proof Take X and V_α matrices with orthonormal columns such that $\mathcal{X} = \mathrm{Span}(X)$ and $\mathcal{V}_\alpha = \mathrm{Span}(V_\alpha)$. Since $\theta_k < \pi/2$, we know that $p = 0$ in Lemma 2.5 and thus $s = k - r$ with r the number of principal angles that are equal to zero. Choosing a matrix $X_\perp$ with orthonormal columns such that $\mathrm{Span}(X_\perp) = \mathrm{Span}(X)^\perp$, we therefore get from Lemma 2.5 that there exist orthogonal matrices U_1, V_1 of size k and V_2 of size $n - k$ such that

$$V_\alpha^T X = U_1 \begin{bmatrix} I_r & \\ & C_{k-r} \end{bmatrix} V_1^T, \qquad V_\alpha^T X_\perp = U_1 \begin{bmatrix} O_{r\times m} & \\ & S_{k-r} \end{bmatrix} V_2^T. \tag{2.2.2.3}$$

Comparing with Definition 1.9, we deduce that $C_{k-r} = \mathrm{diag}(\cos\theta_{r+1}, \ldots, \cos\theta_k)$ and $S_{k-r} = \mathrm{diag}(\sin\theta_{r+1}, \ldots, \sin\theta_k)$ since $C_{k-r}^2 + S_{k-r}^2 = I$.

We recall from (1.3.1.6) that

$$\mathrm{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) = U\,\mathrm{atan}(\Sigma)V^T, \tag{2.2.2.4}$$

where $U\Sigma V^T = (I_n - XX^T)V_\alpha(X^T V_\alpha)^{-1} =: M$ is a compact SVD (without the requirement that the diagonal of Σ is non-increasing). Using $X_\perp$ from above, we can also write $M = X_\perp X_\perp^T V_\alpha (X^T V_\alpha)^{-1}$. Substituting (2.2.2.3) and using that U_1 and V_1 are orthogonal gives

$$M = X_\perp V_2 \begin{bmatrix} O_{m\times r} & \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix} V_1^T = X_\perp \tilde{V}_2 \begin{bmatrix} O_{r\times r} & \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix} V_1^T,$$

where $\tilde{V}_2 \in \mathbb{R}^{(n-k)\times k}$ contains the last k columns of V_2 in order. Note that this reformulation of the SVD of M holds always, regardless of the relationship between m and r. If $m \geq r$, the matrix $\begin{bmatrix} O_{m\times r} & \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix}$ has its first $m - r$ rows equal to 0, thus we can cut the first $m - r$ columns of V_2, since they do not contribute to the product. This yields a matrix $\tilde{V}_2$ with $n - k$ rows and

$n-k-m+r=k$ of the last columns of V_2. If $m<r$, then the first $r-m$ columns of $\begin{bmatrix} O_{m\times r} \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix}$ are 0 and now we can add $r-m$ columns in the beginning of the matrix V_2 that keep the derived matrix orthonormal. This again yields a matrix $\tilde{V}_2$ with $n-k$ rows and $n-k+r-m=k$ columns. Since the matrix $\begin{bmatrix} O_{r\times r} \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix}$ occurs by adding $r-m$ zero rows at the beginning of $\begin{bmatrix} O_{m\times r} \\ & S_{k-r}C_{k-r}^{-1} \end{bmatrix}$, the product does not change.

Since $\theta_1=\cdots=\theta_r=0$, we can therefore formulate the compact SVD of M using the vector θ of all principal angles as follows:

$$M=U\Sigma V^T \quad \text{with } U=X_\perp\tilde{V}_2,\ \Sigma=\tan(\theta),\ V=V_1.$$

Hence from (2.2.2.4) we get directly that

$$\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)=X_\perp\tilde{V}_2\,\theta\,V_1^T, \tag{2.2.2.5}$$

where θ is a diagonal matrix.

We now claim that (2.2.2.5) also satisfies

$$\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)=X_\perp X_\perp^T V_\alpha U_1\frac{\theta}{\sin\theta}V_1^T, \tag{2.2.2.6}$$

where $\frac{\theta}{\sin\theta}$ is a diagonal matrix for which $\frac{0}{\sin 0}=1$. Indeed, recalling that $\theta_1=\cdots=\theta_r=0$ and using the identities

$$X_\perp^TV_\alpha=\tilde{V}_2\begin{bmatrix} O_{r\times r} \\ & S_{k-r}\end{bmatrix}U_1^T,\quad \frac{\theta}{\sin\theta}=\begin{bmatrix} I_r \\ & S_{k-r}^{-1}\end{bmatrix}\begin{bmatrix} I_r \\ & T_{k-r}\end{bmatrix}$$

where $T_{k-r}=\operatorname{diag}(\theta_{r+1},\ldots,\theta_k)$, we obtain

$$\begin{aligned}\text{RHS of (2.2.2.6)} &= X_\perp\tilde{V}_2\begin{bmatrix} O_{r\times r} \\ & S_{k-r}\end{bmatrix}\begin{bmatrix} I_r \\ & S_{k-r}^{-1}\end{bmatrix}\begin{bmatrix} I_r \\ & T_{k-r}\end{bmatrix}V_1^T \\ &= X_\perp\tilde{V}_2\begin{bmatrix} O_{r\times r} \\ & T_{k-r}\end{bmatrix}V_1^T = X_\perp\tilde{V}_2\,\theta\,V_1^T = \text{RHS of (2.2.2.5)}.\end{aligned}$$

Next, we work out

$$s:=\langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)\rangle.$$

Since $\operatorname{grad} f(\mathcal{X})$ and $\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)$, respectively, give tangent vectors for the same representative X of $\mathcal{X}$, the inner product above is the trace of the corresponding matrix representations. Using (2.2.2.6) with $I-XX^T=X_\perp X_\perp^T$, we therefore get

$$\begin{aligned}s &= 2\Big\langle (I-XX^T)AX, (I-XX^T)V_\alpha U_1\frac{\theta}{\sin(\theta)}V_1^T\Big\rangle \\ &= 2\operatorname{Tr}\Big(\frac{\theta}{\sin(\theta)}U_1^TV_\alpha^T(I-XX^T)AXV_1\Big).\end{aligned}$$

Since $AV_\alpha = V_\alpha \Lambda_\alpha$, we can simplify

$$V_\alpha^T (I - XX^T)AX = \Lambda_\alpha V_\alpha^T X - V_\alpha^T XX^T AX. \tag{2.2.2.7}$$

Substituting in the expression above and using that $V_\alpha^T X = U_1 \cos\theta \, V_1^T$, we get

$$\begin{aligned}\frac{1}{2}s &= \operatorname{Tr}\Big(\frac{\theta}{\sin(\theta)} U_1^T \Lambda_\alpha U_1 \cos(\theta)\Big) - \operatorname{Tr}\Big(\frac{\theta}{\sin(\theta)}\cos(\theta) V_1^T X^T AXV_1\Big) \\ &= \operatorname{Tr}\Big(\frac{\theta}{\tan(\theta)}\Big(U_1^T \Lambda_\alpha U_1 - V_1^T X^T AXV_1\Big)\Big),\end{aligned}$$

with the convention $\frac{0}{\tan 0} = 1$.

Denote the symmetric matrix

$$S := U_1^T \Lambda_\alpha U_1 - V_1^T X^T AXV_1. \tag{2.2.2.8}$$

We show below that all diagonal entries $S_{11}, \ldots, S_{kk}$ of S are nonnegative. Hence, by diagonality of the matrix $\frac{\theta}{\tan(\theta)}$, we obtain

$$\frac{1}{2}s = \sum_j \frac{\theta_j}{\tan\theta_j} S_{jj} \geq \min_j \frac{\theta_j}{\tan\theta_j} \operatorname{Tr}(S) = \frac{\theta_k}{\tan\theta_k}\Big[\operatorname{Tr}(\Lambda_\alpha) - \operatorname{Tr}(X^T AX)\Big]$$

since U_1 and V_1 are orthogonal matrices. We recover the desired result after substituting $f(\mathcal{X}) = -\operatorname{Tr}(X^T AX)$ and $f^* = -\operatorname{Tr}(V_\alpha^T AV_\alpha) = -\operatorname{Tr}(\Lambda_\alpha)$.

It remains to show that $S_{jj} \geq 0$ for $j = 1, \ldots, k$. Since $\operatorname{Span}(V_\beta) = \operatorname{Span}(V_\alpha)^\perp$, Lemma 2.5 gives us in addition to (2.2.2.3) also

$$V_\beta^T X = U_2 \begin{bmatrix} O_{m\times r} \\ & S_{k-r} \end{bmatrix} V_1^T = \tilde{U}_2 \sin\theta \, V_1^T, \tag{2.2.2.9}$$

where $\tilde{U}_2 \in \mathbb{R}^{(n-k)\times k}$ contains the last k columns of the orthogonal matrix U_2 in order. A short calculation using (2.2.2.1) then shows that (2.2.2.8) satisfies

$$S = U_1^T \Lambda_\alpha U_1 - \cos\theta \, U_1^T \Lambda_\alpha U_1 \cos\theta - \sin\theta \, \tilde{U}_2^T \Lambda_\beta \tilde{U}_2 \sin\theta$$

with diagonal elements

$$S_{jj} = \sin^2\theta_j \, (U_1^T \Lambda_\alpha U_1 - \tilde{U}_2^T \Lambda_\beta \tilde{U}_2)_{jj}.$$

Since U_1 and $\tilde{U}_2$ have orthonormal columns, we obtain

$$\lambda_{\min}(U_1^T \Lambda_\alpha U_1) \geq \lambda_{\min}(\Lambda_\alpha) = \lambda_k, \quad \lambda_{\max}(\tilde{U}_2^T \Lambda_\beta \tilde{U}_2) \leq \lambda_{\max}(\Lambda_\beta) = \lambda_{k+1},$$

from which we get with Weyl's inequality that

$$\lambda_{\min}(U_1^T \Lambda_\alpha U_1 - \tilde{U}_2^T \Lambda_\beta \tilde{U}_2) \geq \lambda_{\min}(U_1^T \Lambda_\alpha U_1) - \lambda_{\max}(\tilde{U}_2^T \Lambda_\beta \tilde{U}_2) \geq \lambda_k - \lambda_{k+1} \geq 0.$$

Hence, the matrix

$$U_1^T \Lambda_\alpha U_1 - \tilde{U}_2^T \Lambda_\beta \tilde{U}_2 \tag{2.2.2.10}$$

is symmetric and positive semi-definite. Its diagonal entries, and thus also S_{jj}, are therefore nonnegative. ■

We are finally able to show the promised WQSC property of the symmetric eigenvalue problem (recall Definition 1.22).

Theorem 2.7 (Weak-quasi-strong convexity) *Let $0 \leq \theta_1 \leq \cdots \leq \theta_k < \pi/2$ be the principal angles between the subspaces $\mathcal{X}$ and $\mathcal{V}_\alpha$. Then, f satisfies*

$$f(\mathcal{X}) - f^* \leq \frac{1}{a(\mathcal{X})} \langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) \rangle - c_Q \delta \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha)$$

with $a(\mathcal{X}) = \theta_k / \tan\theta_k > 0$, $c_Q = 4/\pi^2 > 0.4$, and $\delta = \lambda_k - \lambda_{k+1} \geq 0$.

Proof Combining Propositions 3.2 and 2.6 leads to

$$c_Q \delta \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha) \leq f(\mathcal{X}) - f^* \leq \frac{1}{2a(\mathcal{X})} \langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) \rangle.$$

At the same time, Proposition 2.6 also implies

$$\begin{aligned} f(\mathcal{X}) - f^* \leq \frac{1}{2a(\mathcal{X})} \langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) \rangle - c_Q \delta \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha) \\ + c_Q \delta \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha). \end{aligned}$$

Using the first inequality to bound the last term of the right hand side, we recover the desired result. ■

Remark 2.1 *Theorem 2.7 is also valid when the spectral gap $\delta = 0$. In that case, $\mathcal{V}_\alpha$ is any subspace spanned by k leading eigenvectors of A and the theorem (almost) reduces to Proposition 2.6 (up to a scalar 2).*

While not needed for our convergence proof, the next result is of independent interest and shows that f is PL in the Riemannian sense when the spectral gap δ is strictly positive. This property generalizes a result by [117] for the Rayleigh quotient in the sphere.

Proposition 2.8 (PL condition) *The function f satisfies*

$$\|\operatorname{grad} f(\mathcal{X})\|^2 \geq 4\, c_Q\, \delta\, a^2(\mathcal{X}) (f(\mathcal{X}) - f^*)$$

for all subspaces $\mathcal{X}$ that have a largest principal angle $< \pi/2$ with $\mathcal{V}_\alpha$.

Proof We assume that $\delta > 0$ since otherwise the statement is trivially true. By Theorem 2.7, we have

$$f(\mathcal{X}) - f^* \leq \frac{1}{a(\mathcal{X})} \langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) \rangle - c_Q \delta \operatorname{dist}^2(\mathcal{X}, \mathcal{V}_\alpha).$$

Since $\langle G_1, G_2 \rangle \leq \frac{\rho}{2}\|G_1\|^2 + \frac{1}{2\rho}\|G_2\|^2$ for all matrices G_1, G_2 and $\rho > 0$, we can write (for any $\rho > 0$) that

$$\langle \operatorname{grad} f(\mathcal{X}), -\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha) \rangle \leq \frac{\rho}{2}\|\operatorname{grad} f(\mathcal{X})\|^2 + \frac{1}{2\rho}\|\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)\|^2.$$

Using that $\operatorname{dist}(\mathcal{X}, \mathcal{V}_\alpha) = \|\operatorname{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)\|$ and choosing $\rho = 1/(2c_Q \delta a(\mathcal{X}))$, we get the desired result. ∎

2.3 Convergence of Riemannian gradient descent

We now have everything in place to prove the convergence of the Riemannian gradient descent (RGD) method on the Grassmann manifold for minimizing f. Starting from a subspace $\mathcal{X}_0 \in \operatorname{Gr}(n, k)$, we iterate

$$\mathcal{X}_{t+1} = \operatorname{Exp}_{\mathcal{X}_t}(-\eta_t \operatorname{grad} f(\mathcal{X}_t)). \tag{2.3.0.1}$$

Here, $\eta_t > 0$ is a step size that may depend on the iteration t and will be carefully chosen depending on the specific case, but always depending on L, which equals $2(\lambda_1 - \lambda_n)$.

We start by a general result which shows that the distance to the optimal subspace contracts after one step of gradient descent. The step size depends on the smoothness and weak-quasi convexity constants of f from Propositions 2.1 and 2.6. This is crucial since the constant $a(\mathcal{X})$ depends on the biggest principal angle between $\mathcal{X}$ and $\mathcal{V}_\alpha$ and bounding the evolution of distances of the iterates to the minimizer will help us also bound this constant[6]. An alternative contraction property with a more tractable step size is presented in Proposition 2.15 of Section 2.4.

Lemma 2.9 (Contraction of RGD) *Let $\mathcal{X}_t$ and $\mathcal{V}_\alpha$ have principal angles $0 \leq \theta_1 \leq \cdots \leq \theta_k < \pi/2$. Then, iteration* (2.3.0.1) *with $0 \leq \eta_t \leq a(\mathcal{X}_t)/L$ satisfies*

$$\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq \big(1 - 2c_Q \delta a(\mathcal{X}_t)\, \eta_t\big) \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha).$$

Observe that $L = 0$ implies $A = \lambda_1 I$ and any subspace $\mathcal{X}$ of dimension k will be an eigenspace of A with $\operatorname{dist}(\mathcal{X}, \mathcal{V}_\alpha) = 0$. We will therefore not explicitly

[6]The analysis of [50] is wrong with respect to this issue as discussed in detail in [8].

prove this lemma and all forthcoming convergence results for $L = 0$ since the statements will be trivially true.

Proof [Proof of Lemma 2.9] By the assumption on the principal angles, we get that $0 < a(\mathcal{X}_t) = \theta_k / \tan\theta_k \leq 1$. The hypothesis on η_t and Lemma 2.2 then gives

$$\eta_t \| \operatorname{grad} f(\mathcal{X}_t)\|_2 \leq \frac{a(\mathcal{X}_t)}{L} \| \operatorname{grad} f(\mathcal{X}_t)\|_2 \leq \frac{1}{2} < \frac{\pi}{2}.$$

By (1.3.1.5), this guarantees that the geodesic $\tau \mapsto \operatorname{Exp}(-\tau\eta_t \operatorname{grad} f(\mathcal{X}_t))$ lies within the injectivity domain at $\mathcal{X}_t$ for $\tau \in [0, 1]$. Hence, Exp is bijective along this geodesic and thus $\operatorname{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1}) = -\eta_t \operatorname{grad} f(\mathcal{X}_t)$. We can thus apply Proposition 1.15 to obtain

$$\begin{aligned} \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) &\leq \| -\eta_t \operatorname{grad} f(\mathcal{X}_t) - \operatorname{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\|^2 \\ &= \eta_t^2 \| \operatorname{grad} f(\mathcal{X}_t)\|^2 + \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) + 2\eta_t \sigma \end{aligned} \tag{2.3.0.2}$$

with

$$\sigma := \langle \operatorname{grad} f(\mathcal{X}_t), \operatorname{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle.$$

Theorem 2.7 and Lemma 2.3 together with Proposition 2.1 (see also Proposition 1.21) give

$$\begin{aligned} \frac{\sigma}{a(\mathcal{X}_t)} &\leq f^* - f(\mathcal{X}_t) - c_Q \delta \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \\ &\leq -\frac{1}{2L} \| \operatorname{grad} f(\mathcal{X}_t)\|^2 - c_Q \delta \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha). \end{aligned}$$

Multiplying by $2a(\mathcal{X}_t)\,\eta_t$ and using $\eta_t \leq a(\mathcal{X}_t)/L$, we get

$$\begin{aligned} 2\eta_t \sigma &\leq -\frac{a(\mathcal{X}_t)\,\eta_t}{L} \| \operatorname{grad} f(\mathcal{X}_t)\|^2 - 2c_Q \delta a(\mathcal{X}_t)\,\eta_t \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \\ &\leq -\eta_t^2 \| \operatorname{grad} f(\mathcal{X}_t)\|^2 - 2c_Q \delta a(\mathcal{X}_t)\,\eta_t \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha). \end{aligned}$$

Substituting into (2.3.0.2), we obtain the first statement of the lemma. ■

Remark 2.2 *When $\delta = 0$, Lemma 2.9 still holds* for any subspace $\mathcal{V}_\alpha$ spanned by k leading eigenvectors of A. *In that case, the lemma only guarantees that the distance between the iterates of gradient descent and this $\mathcal{V}_\alpha$ does not increase.*

2.3.1 Linear convergence rate under positive spectral gap

Lemma 2.9 features a contraction rate only for one step of the algorithm. In order to get a global convergence rate, one needs to bound the quantity $a(\mathcal{X}_t)$ from below and independently of t. To that end, we need a stricter bound in the distance of the initial guess to the optimum. Such a bound guarantees that $a(X_t)$ remains always lower bounded by a positive number, or equivalently, that the iterates of the algorithm never get too close to a non-optimal critical point.

Theorem 2.10 *If* $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ *then the iterates* $\mathcal{X}_t$ *of Riemannian gradient descent* (2.3.0.1) *with step size* η_t *such that*

$$0 < \eta \leq \eta_t \leq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))/L$$

satisfy

$$\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \leq (1 - 2c_Q \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\, \delta\, \eta)^t \operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha).$$

Proof We first claim that $\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$ for all $t \geq 0$. This would then also imply that $\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha) < \pi/2$ for all $t \geq 0$ since

$$\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \sqrt{\sum_{i=1}^{k} \theta_i(\mathcal{X}_t, \mathcal{V}_\alpha)^2} = \operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha).$$

For $t = 0$, we have $\theta_k(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ by hypothesis on $\mathcal{X}_0$ and thus

$$a(\mathcal{X}_0) = \frac{\theta_k(\mathcal{X}_0, \mathcal{V}_\alpha)}{\tan(\theta_k(\mathcal{X}_0, \mathcal{V}_\alpha))} \geq \cos(\theta_k(\mathcal{X}_0, \mathcal{V}_\alpha)) \geq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)).$$

Since by construction $\eta_0 \leq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))/L$, this implies that $\eta_0 \leq a(\mathcal{X}_0)/L$ and Lemma 2.9 guarantees that $\operatorname{dist}(\mathcal{X}_1, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$. In particular, we also have $\theta_k(\mathcal{X}_1, \mathcal{V}_\alpha) < \pi/2$.

Next, assume that

$$\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha),$$

which implies $\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha) < \pi/2$. Then by a similar argument like above, we have

$$a(\mathcal{X}_t) \geq \cos(\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha)) \geq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)). \tag{2.3.1.1}$$

By hypothesis on η_t, we observe

$$\eta_t \leq \frac{\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}{L} \leq \frac{\cos(\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha))}{L} \leq \frac{a(\mathcal{X}_t)}{L}.$$

Applying Lemma 2.9 once again with the induction hypothesis proves the claim:

$$\operatorname{dist}(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha).$$

The main statement of the theorem now follows easily: Since $\eta_t \leq a(\mathcal{X}_t)/L$ and $\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha) < \pi/2$ for all $t \geq 0$, Lemma 2.9 gives

$$\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq (1 - 2c_Q a(\mathcal{X}_t)\delta\eta_t) \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha).$$

Combining with (2.3.1.1) and $\eta_t \geq \eta$ shows the desired result by induction. ■

If the spectral gap δ is strictly positive, then Theorem 2.10 gives an exponential convergence rate towards the optimum $\mathcal{V}_\alpha$. If $\delta = 0$, then Theorem 2.10 *does not provide a convergence rate* but rather implies that the intrinsic distances of the iterates to the optimum do not increase.

From Theorem 2.10 we get immediately the following iteration complexity.

Corollary 2.11 *Let Riemannian gradient descent starting from a subspace* $\mathcal{X}_0$ *that satisfies* $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ *and with step size* η *satisfying the condition of Theorem* 2.10*. Then after at most*

$$T = 2\frac{\log(\varepsilon) - \log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}{\log(1 - 0.8\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta)} + 1 \leq \mathcal{O}\left(\frac{\log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)) - \log(\varepsilon)}{\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta}\right)$$

many iterations, $\mathcal{X}_T$ *will satisfy* $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \varepsilon$*. With the maximal step size allowed in Theorem* 2.10*, we get*

$$T \leq \mathcal{O}\left(\frac{\lambda_1 - \lambda_n}{\delta}\frac{1}{\cos^2(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}\operatorname{Log}\left(\frac{\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)}{\varepsilon}\right)\right).$$

Proof In order to guarantee $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$, it suffices to have

$$(1 - 2c_Q\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\,\delta\,\eta)^T \operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \epsilon^2.$$

Taking the logarithm of both sides, we get

$$T\log(1 - 2c_Q\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta) + 2\log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)) \leq 2\log(\epsilon),$$

which gives

$$T \geq 2\frac{\log(\epsilon) - \log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}{\log(1 - 2c_Q\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))},$$

since $\log(1 - 2c_Q\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))$ is negative. By considering that $c_Q \geq 0.8$, we get

$$T \geq 2\frac{\log(\epsilon) - \log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}{\log(1 - 0.8\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta)},$$

and the smallest integer that satisfies this inequality is exactly

$$T = 2\frac{\log(\epsilon) - \log(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))}{\log(1 - 0.8\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta)}.$$

The inequality part of the result follows by considering that

$$\log(1 - 0.8\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)\delta\eta) \geq -\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))\delta\eta.$$

The final bound for T follows by a simple substitution of $\eta = \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))/L$. ■

As expected, T depends inversely proportional on the spectral gap δ and proportional to the spread of the eigenvalues. In addition, we also have an extra term $1/\cos^2(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))$ that depends on the initial distance $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$, which is due to the weak-quasi convexity property of f. This is a conservative overestimation, since this quantity improves as the iterates get closer to the optimum.

Remark 2.3 *If $\delta > 0$, the exponential convergence rate is in terms of the intrinsic distance on the Grassmann manifold, that is, the ℓ_2 norm of the principal angles. Standard convergence results for subspace iteration are stated for the biggest principal angle, that is, the ℓ_∞ norm. This is weaker than the intrinsic distance. For subspace iteration with projection, the convergence result from [96, Thm. 5.2] shows that all principal angles θ_i converge to zero and eventually gives convergence of the ℓ_4 norm of the principal angles. This is also weaker than the intrinsic distance.*

2.3.2 Convergence of function values without a spectral gap assumption

When $\delta = 0$, Theorem 2.10 still holds, but does not provide a rate of convergence as discussed above. Instead, we can prove the following result:

Theorem 2.12 *If the distance* $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$ *of the initial subspace* $\mathcal{X}_0$ *to the minimizer satisfies* $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ *for a subspace* $\mathcal{V}_\alpha$ *that is spanned by any k leading eigenvectors of A, then the iterates $\mathcal{X}_t$ of Riemannian gradient descent* (2.3.0.1) *with fixed step size*

$$\eta \leq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))/L$$

satisfy

$$f(\mathcal{X}_t) - f^* \leq \frac{2L + \frac{1}{\eta}}{4(\cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))t + 1)} \operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) = \mathcal{O}\left(\frac{1}{t}\right).$$

Proof Since we satisfy all the hypotheses of Theorem 2.10, we know that for all $t \geq 0$ it holds $\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ and thus also that $\mathcal{X}_t$ is in the injectivity domain of Exp at $\mathcal{V}_\alpha$. In addition, its proof states in (2.3.1.1) that

$$a(\mathcal{X}_t) \geq C_0 := \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)) > 0,$$

which implies that the function f is weakly-quasi-convex at every $\mathcal{X}_t$ with constant $2C_0$. Hence

$$2C_0\Delta_t \leq \langle \operatorname{grad} f(\mathcal{X}_t), -\operatorname{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle, \tag{2.3.2.1}$$

where we defined

$$\Delta_t := f(\mathcal{X}_t) - f^*.$$

Similar to the proof of Theorem 2.10, by the hypothesis on the step size η_t, Lemma 2.9 shows that $\mathcal{X}_{t+1}$ is in the injectivity domain of Exp at $\mathcal{X}_t$. Hence, by the definition of Riemannian gradient descent, we have

$$\operatorname{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1}) = -\eta \operatorname{grad} f(\mathcal{X}_t). \tag{2.3.2.2}$$

In addition, the smoothness property (2.2.1.1) of f gives

$$\Delta_{t+1} - \Delta_t \leq \langle \operatorname{grad} f(\mathcal{X}_t), \operatorname{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1})\rangle + \frac{L}{2} \operatorname{dist}^2(\mathcal{X}_t, \mathcal{X}_{t+1}).$$

Substituting (2.3.2.2), we obtain

$$\Delta_{t+1} - \Delta_t \leq \left(-\eta + \frac{L}{2}\eta^2\right) \|\operatorname{grad} f(\mathcal{X}_t)\|^2 \leq 0, \tag{2.3.2.3}$$

since $\eta \leq C_0/L$ with $0 < C_0 := \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)) \leq 1$ and $L > 0$.

Since $\operatorname{Gr}(n,k)$ has nonnegative sectional curvature, Proposition 1.15 implies

$$\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq \operatorname{dist}^2(\mathcal{X}_t, \mathcal{X}_{t+1}) + \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - 2\langle \operatorname{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1}), \operatorname{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle.$$

Substituting (2.3.2.2) into the above and rearranging terms gives

$$2\eta\langle \operatorname{grad} f(\mathcal{X}_t), -\operatorname{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle \leq \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) + \eta^2 \|\operatorname{grad} f(\mathcal{X}_t)\|^2.$$

Combining with (2.3.2.1), we get

$$\Delta_t \leq \frac{1}{4C_0\eta}(\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)) + \frac{\eta}{4C_0}\|\operatorname{grad} f(\mathcal{X}_t)\|^2. \tag{2.3.2.4}$$

Now multiplying (2.3.2.3) by $\frac{1}{C_0}$ and summing with (2.3.2.4) gives

$$\begin{aligned}\frac{1}{C_0}\Delta_{t+1} - \left(\frac{1}{C_0} - 1\right)\Delta_t \leq{}& \frac{1}{4C_0\eta}(\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)) \\ &+ \frac{1}{C_0}\left(-\eta + \frac{L}{2}\eta^2 + \frac{\eta}{4}\right)\|\operatorname{grad} f(\mathcal{X}_t)\|^2. \end{aligned} \tag{2.3.2.5}$$

By assumption $\eta \leq C_0/L$, where $0 < C_0 := \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)) \leq 1$ and $L > 0$. Since

$$\frac{\eta}{C_0}\left(-1 + \frac{L}{2}\eta + \frac{1}{4}\right) \leq \frac{\eta}{C_0}\left(\frac{C_0}{2} - \frac{3}{4}\right) \leq -\frac{1}{4}\frac{\eta}{C_0} < 0.$$

Inequality (2.3.2.5) can be simplified to

$$\frac{1}{C_0}\Delta_{t+1} - \left(\frac{1}{C_0} - 1\right)\Delta_t \leq \frac{1}{4C_0\eta}(\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)).$$

Summing from 0 to $t-1$ gives

$$\frac{1}{C_0}\Delta_t + \sum_{s=1}^{t-1}\Delta_s - \left(\frac{1}{C_0} - 1\right)\Delta_0 \leq \frac{1}{4C_0\eta}\left(\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha)\right).$$

From the smoothness property (2.2.1.1) at the critical point $\mathcal{V}_\alpha$ of f, we get

$$\Delta_0 \leq \frac{L}{2}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha).$$

Combining these two inequalities then leads to

$$\frac{1}{C_0}\Delta_t + \sum_{s=0}^{t-1}\Delta_s \leq \frac{1}{C_0}\Delta_0 + \frac{1}{4C_0\eta}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)$$
$$\leq \frac{1}{2C_0}\left(L + \frac{1}{2\eta}\right)\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha).$$

Since (2.3.2.3) holds for all $t \geq 0$, it also implies $\Delta_t \leq \Delta_s$ for all $1 \leq s \leq t$. Substituting

$$t\Delta_t \leq \sum_{s=0}^{t-1}\Delta_s$$

into the inequality from above,

$$\Delta_t \leq \frac{1}{2C_0}\frac{L + \frac{1}{2\eta}}{\frac{1}{C_0} + t}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) = \frac{L + \frac{1}{2\eta}}{2(C_0 t + 1)}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha),$$

we obtain the desired result. ■

Remark 2.4 *This type of result is standard for functions that are geodesically convex (see, e.g. [118]). Our objective function does not satisfy this property, but we can still have a similar upper bound on the iteration complexity for convergence in function value. We note that this does not imply convergence of the iterates to a specific k-dimensional subspace, but only convergence of a subsequence of the sequence of the iterates.*

2.3.3 Sufficiently small step sizes

The convergence results in Theorems 2.10 and 2.12 require that the initial subspace $\mathcal{X}_0$ lies within a distance strictly less than $\pi/2$ from a global minimizer $\mathcal{V}_\alpha$. While this condition is independent from the spectral gap (unlike results that rely on standard convexity, see Section 2.5), it is also not fully satisfactory: it is hard to verify in practice, and it is unnecessarily severe in numerical experiments. In fact, this condition is only used to obtain a uniform lower bound on the weak-quasi convexity constant $a(\mathcal{X}_t) = \theta_k^{(t)}/\tan(\theta_k^{(t)})$ with $\theta_k^{(t)}$ the largest principal angle between $\mathcal{X}_t$ and $\mathcal{V}_\alpha$. Since the Riemannian distance is the ℓ_2 norm of the principal angles, a contraction in this distance leads automatically to $\theta_k^{(t)} < \pi/2$ if $\theta_k^{(0)} < \pi/2$. If one could guarantee by some other reasoning that $\theta_k^{(t)}$ does not increase after one step, the condition $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) < \pi/2$ would not be needed.

We now show that for sufficiently small step sizes η_t, the largest principal angle $\theta_k^{(t)}$ between $\mathcal{X}_t$ and $\mathcal{V}_\alpha$ does indeed not increase after each iteration of

Riemannian gradient descent regardless of the initial subspace $\mathcal{X}_0$. While it does not explain what we observe in numerical experiments where large steps can be taken, it is a first result in explaining why we can initialize the iteration at a random initial subspace $\mathcal{X}_0$.

Proposition 2.13 *Riemannian gradient descent started from a subspace $\mathcal{X}_t$ returns a subspace $\mathcal{X}_{t+1}$ such that*

$$\theta_k(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq \theta_k(\mathcal{X}_t, \mathcal{V}_\alpha),$$

for all step sizes $0 \leq \eta \leq \bar{\eta}$ where $\bar{\eta} > 0$ is sufficiently small.

For the proof of this proposition, we will need the derivatives of certain singular values. While this is well known for isolated singular values, it is possible to generalize to higher multiplicities as well by relaxing the ordering and sign of singular values [27]. For a concrete formula, we use the following result from Lemma A.5 in [72].

Lemma 2.14 *Let $\sigma_1 \geq \cdots \geq \sigma_n$ be the singular values of $S \in \mathbb{R}^{n\times n}$ with $u_1, \ldots, u_n$ and $v_1, \ldots, v_p$ the associated left and right orthonormal singular vectors. Suppose that σ_j has multiplicity m, that is,*

$$\sigma_{j_0-1} > \sigma_{j_0} = \cdots = \sigma_j = \cdots = \sigma_{j_0+m-1} > \sigma_{j_0+m}.$$

Then, the jth singular value of $S + \eta T$ satisfies

$$\sigma_j(S + \eta T) = \sigma_j + \eta\lambda_{j-j_0+1} + \mathcal{O}(\eta^2), \quad \eta \to 0^+,$$

where λ_j is the jth largest eigenvalue of $\frac{1}{2}(U^T B V + V^T B^T U)$ with

$$U = \begin{bmatrix} u_{j_0} & \cdots & u_{j_0+m-1} \end{bmatrix} \quad \text{and} \quad V = \begin{bmatrix} v_{j_0} & \cdots & v_{j_0+m-1} \end{bmatrix}.$$

Proof [Proof of Proposition 2.13]. For ease of notation, let $X := X_t$ and $X_+ := X_{t+1}$ such that $\mathcal{X}_t = \mathrm{Span}(X)$ and $\mathcal{X}_{t+1} = \mathrm{Span}(X_+)$. By definition of the exponential map on Grassmann, the next iterate of the Riemannian GD iteration (2.3.0.1) with step η satisfies

$$X_+ = XV\cos(\eta\Sigma)V^T + U\sin(\eta\Sigma)V^T$$

where

$$U\Sigma V^T = -\operatorname{grad} f(\mathcal{X}_t).$$

Since V is orthogonal, we can write

$$U\sin(\eta\Sigma)V^T = U(\eta\Sigma)V^T V\left(\frac{\sin(\eta\Sigma)}{\eta\Sigma}\right)V^T = -\eta\operatorname{grad} f(\mathcal{X}_t)V\left(\frac{\sin(\eta\Sigma)}{\eta\Sigma}\right)V^T$$

where $1/\Sigma := \Sigma^{-1}$ and $\frac{\sin 0}{0} = 1$. Taking Taylor expansions of sin and cos,

$$V\cos(\eta\Sigma)V^T = V\left(I - \mathcal{O}(\eta^2)\right)V^T = I - \mathcal{O}(\eta^2)$$
$$V\frac{\sin(\eta\Sigma)}{\eta\Sigma}V^T = V\left(I - \mathcal{O}(\eta^2)\right)V^T = I - \mathcal{O}(\eta^2),$$

we obtain

$$\begin{aligned} V_\alpha^T X_+ &= V_\alpha^T X(I - \mathcal{O}(\eta^2)) + V_\alpha^T(-\eta \operatorname{grad} f(\mathcal{X}))(I - \mathcal{O}(\eta^2)) \\ &= V_\alpha^T(X - \eta \operatorname{grad} f(\mathcal{X}_t))(I - \mathcal{O}(\eta^2)) \end{aligned} \tag{2.3.3.1}$$

since $\|V_\alpha\|_2 = \|X\|_2 = 1$.

Let now θ be the vector of k principal angles between $\mathcal{X}_t$ and $\mathcal{V}_\alpha$. As in (2.2.2.2) and (2.2.2.9), we therefore have the SVDs

$$V_\alpha^T X = U_1 \cos\theta\, V_1^T \qquad \text{and} \qquad V_\beta^T X = \tilde{U}_2 \sin\theta\, V_1^T, \tag{2.3.3.2}$$

where $U_1, V_1 \in \mathbb{R}^{k\times k}$ and $\tilde{U}_2 \in \mathbb{R}^{(n-k)\times k}$ have orthonormal columns. Next, we write (2.3.3.1) in terms of

$$M := \sin^2\theta\, U_1^T \Lambda_\alpha U_1 \cos\theta - \cos\theta \sin\theta\, \tilde{U}_2^T \Lambda_\beta \tilde{U}_2 \sin\theta.$$

Since $\operatorname{grad} f(\mathcal{X}_t) = -2(I - XX^T)AX$, the identity (2.2.2.7) gives

$$V_\alpha^T(X - \eta \operatorname{grad} f(\mathcal{X}_t)) = V_\alpha^T X + 2\eta\Lambda_\alpha V_\alpha^T X - 2\eta V_\alpha^T X X^T A X.$$

After substituting (2.2.2.1) and (2.3.3.2), a short calculation using $\cos^2\theta = I - \sin^2\theta$ and the orthogonality of U_1 and V_1 then shows

$$V_\alpha^T(X - \eta \operatorname{grad} f(\mathcal{X}_t)) = U_1(\cos\theta + 2\eta M)V_1^T.$$

Relating back to (2.3.3.1), we thus obtain

$$\begin{aligned} V_\alpha^T X_+ &= U_1(\cos\theta + 2\eta M)V_1^T(I - \mathcal{O}(\eta^2)) \\ &= U_1(\cos\theta + 2\eta M)(I - V_1^T\mathcal{O}(\eta^2)V_1)V_1^T \\ &= U_1(\cos\theta + 2\eta M - \mathcal{O}(\eta^2))V_1^T. \end{aligned}$$

The singular values of $V_\alpha^T X_+$ are therefore the same as the singular values of the matrix $\cos\theta + 2\eta M + \mathcal{O}(\eta^2)$.

By Weyl's inequality (see, e.g., [47, Cor. 7.3.5]), each singular value of $\cos\theta + 2\eta M + \mathcal{O}(\eta^2)$ is $\mathcal{O}(\eta^2)$ close to some singular value of $\cos\theta + 2\eta M$. Let $1 \le j \le k$. Denote the jth singular value of $\cos\theta + 2\eta M$ by $\sigma_j(\eta)$ to which we will apply Lemma 2.14. Let m be the multiplicity of $\sigma_j(0)$. Hence, there exists j_0 such that $\sigma_{j_0}(0) = \cdots = \sigma_j(0) = \cdots = \sigma_{j_0+m-1}(0)$. Since $\cos\theta$ is a diagonal matrix with decreasing diagonal, its ℓth singular value equals $\cos\theta_\ell$ and its associated left/right singular vector is the ℓth canonical vector e_ℓ. Denoting

$$E = \begin{bmatrix} e_{j_0} & \cdots & e_{j_0+m-1} \end{bmatrix},$$

observe that $\cos\theta\, E = \cos\theta_{j_0}\, E$ (here, $\cos\theta$ is a diagonal matrix and $\cos\theta_{j_0}$ is a scalar) and likewise for $\sin\theta\, E$. We thus get

$$E^T M E = \sin^2\theta_{j_0} \cos\theta_{j_0} (U_1^T \Lambda_\alpha U_1 - \tilde{U}_2^T \Lambda_\beta \tilde{U}_2).$$

In the proof of Proposition 2.6, we showed that the matrix in brackets above is symmetric and positive semi-definite (see (2.2.2.10)). Since $0 \leq \theta_{j_0} \leq \pi/2$, the eigenvalues of $E^T M E$ are therefore all non-negative. Lemma 2.14 thus gives that $\sigma_j(\eta) \geq \sigma_j$ for sufficiently small and positive η. Since the singular values of $V_\alpha^T X_+$ are the cosines of the principal angles between $\mathcal{V}_\alpha$ and $\mathcal{X}_{t+1}$ with step size $\eta \geq 0$, we conclude that there exists $\bar{\eta} > 0$ such that for all $\eta \in [0, \bar{\eta}]$ it holds

$$\theta_j(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) \leq \theta_j(\mathcal{X}_t, \mathcal{V}_\alpha).$$

Since j was arbitrary, this finishes the proof. ■

2.4 Convergence with step size $1/L$

We now prove convergence of gradient descent with a more tractable choice of step size compared to the one of Theorem 2.10. However, this requires a slightly better initialization at most $\frac{\pi}{2\sqrt{2}}$ away from the minimizer.

2.4.1 Maximum extent of the iterates

We first prove that gradient descent with step size at most $\frac{1}{L}$ does not guarantee contraction on distances from step to step, but it does guarantees that squares distances at most double over the course of the algorithm:

Proposition 2.15 *Consider gradient descent applied to f with step size $\eta \leq \frac{1}{L}$. If the iterates $\mathcal{X}_t$ satisfy $\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha) < \frac{\pi}{2}$, then they also satisfy*

$$\mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \leq 2\mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha).$$

Proof Consider the discrete Lyapunov function

$$\mathcal{E}(t) = \frac{1}{L}(f(\mathcal{X}_t) - f^*) + \frac{1}{2}\mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha).$$

Then

$$\mathcal{E}(t+1) - \mathcal{E}(t) = \frac{1}{L}(f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t)) + \frac{1}{2}(\mathrm{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) - \mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha)).$$

By L-smoothness of f, we have

$$\begin{aligned} f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t) &\leq \langle \mathrm{grad} f(\mathcal{X}_t), \mathrm{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1}) \rangle + \frac{L}{2}\mathrm{dist}(\mathcal{X}_t, \mathcal{X}_{t+1})^2 \\ &= \left(-\eta + \frac{L}{2}\eta^2\right) \|\mathrm{grad} f(\mathcal{X}_t)\|^2. \end{aligned}$$

We also know by Proposition 2.6 that

$$\langle \mathrm{grad} f(\mathcal{X}), -\mathrm{Log}_{\mathcal{X}}(\mathcal{V}_\alpha)\rangle \geq 0,$$

for any $\mathcal{X}$ with $\theta_k(\mathcal{X}, \mathcal{V}_\alpha) < \pi/2$.
By the fact that the sectional curvatures of the Grassmann manifold are non-negative, we have

$$\begin{aligned}
\mathrm{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) &\leq \mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) + \mathrm{dist}^2(\mathcal{X}_{t+1}, \mathcal{X}_t) - 2\langle \mathrm{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1}), \mathrm{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle \\
&= \mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) + \eta^2 \|\mathrm{grad} f(\mathcal{X}_t)\|^2 + 2\eta\langle \mathrm{grad} f(\mathcal{X}_t), \mathrm{Log}_{\mathcal{X}_t}(\mathcal{V}_\alpha)\rangle \\
&\leq \mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) + \eta^2 \|\mathrm{grad} f(\mathcal{X}_t)\|^2.
\end{aligned}$$

Thus

$$\begin{aligned}
\mathcal{E}(t+1) - \mathcal{E}(t) &\leq \left(-\frac{\eta}{L} + \frac{\eta^2}{2}\right) \|\mathrm{grad} f(\mathcal{X}_t)\|^2 + \frac{\eta^2}{2}\|\mathrm{grad} f(\mathcal{X}_t)\|^2 \\
&\leq \left(-\frac{\eta}{L} + \eta^2\right) \|\mathrm{grad} f(\mathcal{X}_t)\|^2 \leq 0,
\end{aligned}$$

because $\eta \leq \frac{1}{L}$.
Since $\mathcal{E}(t)$ does not increase, we have

$$\begin{aligned}
\frac{1}{2}\mathrm{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \mathcal{E}(t) \leq \mathcal{E}(0) &= \frac{1}{L}(f(\mathcal{X}_0) - f^*) + \frac{1}{2}\mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \\
&\leq \frac{1}{2}\mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) + \frac{1}{2}\mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) = \mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)
\end{aligned}$$

and the desired result follows. ■

2.4.2 Convergence under positive spectral gap

When $\delta > 0$, we can use gradient dominance to prove convergence of gradient descent to the (unique) minimizer in terms of function values:

Proposition 2.16 *gradient descent with step size $\eta = 1/L$ initialized at $\mathcal{X}_0$ such that*

$$\mathrm{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{\pi}{4}$$

satisfies

$$f(\mathcal{X}_t) - f^* \leq \left(1 - 0.32 c_Q \frac{\delta}{L}\right)^t (f(\mathcal{X}_0) - f^*).$$

Proof By the previous result and an induction argument to guarantee that the biggest angle between $\mathcal{X}_t$ and $\mathcal{V}_\alpha$ stays strictly less than $\pi/2$, we can bound

the quantities $a(\mathcal{X}_t)$ uniformly from below:
Since $\text{dist}(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \sqrt{2} \cdot \text{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{\sqrt{2}\pi}{4}$, we have

$$a(\mathcal{X}_t) \geq \cos(\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha)) \geq \cos(\text{dist}(\mathcal{X}_t, \mathcal{V}_\alpha)) \geq \cos\left(\frac{\sqrt{2}\pi}{4}\right) \geq 0.4.$$

By L-smoothness of f, we have

$$f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t) \leq -\frac{\|\text{grad} f(\mathcal{X}_t)\|^2}{2L}$$

and applying gradient dominance (Proposition 2.8), we get the bound

$$f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t) \leq -\frac{2c_Q \delta a^2(\mathcal{X}_t)}{L}(f(\mathcal{X}_t) - f^*)$$

thus

$$f(\mathcal{X}_{t+1}) - f^* \leq \left(1 - 2c_Q a^2(\mathcal{X}_t)\frac{\delta}{L}\right)(f(\mathcal{X}_t) - f^*) \leq \left(1 - 0.32c_Q \frac{\delta}{L}\right)(f(\mathcal{X}_t) - f^*).$$

By induction the desired result follows. ■
We now state the iteration complexity of the gradient descent algorithm with step size $\frac{1}{L}$:

Theorem 2.17 *Gradient descent with step size $\frac{1}{L}$ starting from a subspace $\mathcal{X}_0$ with distance at most $\frac{\pi}{4}$ from $\mathcal{V}_\alpha$ computes an estimate $\mathcal{X}_T$ of $\mathcal{V}_\alpha$ such that* $\text{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$ *in at most*

$$T = \frac{1}{0.32c_Q}\frac{L}{\delta}\,\text{Log}\,\frac{f(\mathcal{X}_0) - f^*}{c_Q \delta \epsilon^2} + 1 \leq \mathcal{O}\left(\frac{L}{\delta}\log\frac{f(\mathcal{X}_0) - f^*}{\delta\epsilon}\right).$$

Proof For $\text{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$, it suffices to have

$$f(\mathcal{X}_T) - f^* \leq c_Q \epsilon^2 \delta$$

by quadratic growth of f in Proposition 2.4. Using $(1-c)^T \leq \text{Exp}(-cT)$ for all $T \geq 0$ and $0 \leq c \leq 1$, the previous result gives that it suffices to choose T as the smallest integer such that

$$f(\mathcal{X}_T) - f^* \leq \text{Exp}\left(-0.32c_Q \frac{\delta}{L}T\right)(f(\mathcal{X}_0) - f^*) \leq c_Q \epsilon^2 \delta.$$

Solving for T and substituting $c_Q = 4/\pi^2$, we get the required statement. ■

2.4.3 Gap-less result

We also prove a convergence result for the function values when δ is assumed to be 0:

Theorem 2.18 *Gradient descent with step size $\eta = \frac{1}{L}$ initialized at $\mathcal{X}_0$ such that*

$$\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{\pi}{4}$$

satisfies

$$f(\mathcal{X}_t) - f^* \leq \frac{f(\mathcal{X}_0) - f^* + \frac{L}{2}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)}{0.4t + 1} = \mathcal{O}\left(\frac{1}{t}\right).$$

Proof By Proposition 2.15, we have that $\operatorname{dist}(X_t, \mathcal{V}_\alpha) \leq \frac{\sqrt{2}\pi}{4}$ and f satisfies the weak-quasi convexity inequality at any iterate $\mathcal{X}_t$ of gradient descent with constant $C_0 := 0.4$.

Consider the discrete Lyapunov function

$$\mathcal{E}(t) = \frac{C_0 t + 1}{L}(f(\mathcal{X}_t) - f^*) + \frac{1}{2}\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha).$$

We have that

$$\begin{aligned}\mathcal{E}(t+1) - \mathcal{E}(t) =& \frac{C_0 t + C_0 + 1}{L}(f(\mathcal{X}_{t+1}) - f^*) - \frac{C_0 t + 1}{L}(f(\mathcal{X}_t) - f^*) \\ &+ \frac{1}{2}(\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha)).\end{aligned}$$

Now we have to estimate a bound for $\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha)$. By L-smoothness of f and denoting $\Delta_t = f(\mathcal{X}_t) - f^*$ we have

$$\Delta_{t+1} - \Delta_t \leq \langle \operatorname{grad} f(\mathcal{X}_t), \operatorname{Log}_{\mathcal{X}_t}(\mathcal{X}_{t+1})\rangle + \frac{L}{2}\operatorname{dist}^2(\mathcal{X}_t, \mathcal{X}_{t+1}) = -\frac{\|\operatorname{grad} f(\mathcal{X}_t)\|^2}{2L}$$

By C_0-weak-quasi-strong convexity of f and the fact that the Grassmann manifold is of positive curvature, we have

$$C_0\Delta_t \leq \frac{L}{2}(\operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)) + \frac{\|\operatorname{grad} f(\mathcal{X}_t)\|^2}{2L}$$

Summing this to the previous inequality, we get

$$\operatorname{dist}^2(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) - \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) \leq \frac{2}{L}((1-C_0)(f(\mathcal{X}_t) - f(\mathcal{X}_{t+1})) - C_0(f(\mathcal{X}_{t+1}) - f^*)).$$

Thus

$$\begin{aligned}\mathcal{E}(t+1) - \mathcal{E}(t) &\leq \frac{C_0 t + 1}{L}(f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t)) + \frac{C_0}{L}(f(\mathcal{X}_{t+1}) - f^*) \\ &\quad + \frac{1 - C_0}{L}(f(\mathcal{X}_t) - f(\mathcal{X}_{t+1})) - \frac{C_0}{L}(f(\mathcal{X}_{t+1}) - f^*) \\ &= \frac{C_0 t + C_0}{L}(f(\mathcal{X}_{t+1}) - f(\mathcal{X}_t)) \leq 0.\end{aligned}$$

Thus $\mathcal{E}(t) \leq \mathcal{E}(0)$ and the result follows.

■

2.5 Geodesic convexity

In this section, we show that f is geodesically convex, but only locally around $\mathcal{V}_\alpha$ with a radius that depends on the spectral gap δ. Let $\delta > 0$ and thus $\mathcal{V}_\alpha$ is the unique minimizer of f. Define the following neighbourhood of $\mathcal{V}_\alpha$ in $\mathrm{Gr}(n,k)$:

$$N_*(\varphi) = \{\mathcal{X} \in \mathrm{Gr}(n,k) \colon \theta_k(\mathcal{X}, \mathcal{V}_\alpha) < \varphi\} \qquad \text{with } \varphi \in [0, \pi/4]. \tag{2.5.0.1}$$

Here, $\theta_k(\mathcal{X}, \mathcal{V}_\alpha)$ denotes the largest principal angle between $\mathcal{X}$ and $\mathcal{V}_\alpha$. Since θ_k is a metric on $\mathrm{Gr}(n,k)$ (see [94]), any two subspaces $\mathcal{X}, \mathcal{Y} \in N_*(\varphi)$ will satisfy $\theta_k(\mathcal{X}, \mathcal{Y}) < \pi/2$ by triangle inequality. They thus have a unique connecting geodesic. It is shown in [5, Lemma 2] that for any fixed $\varphi \in [0, \pi/4]$ this geodesic remains in $N_*(\varphi)$. Each set $N_*(\varphi)$ is thus an open totally geodesically convex set as defined in, e.g., [21, Def. 11.16].

One of the main results in [5], namely Cor. 4, states that f is geodesically convex on $N_*(\pi/4)$. This is unfortunately wrong and we present a small counterexample.

Counterexample for Cor. 4 in [5]. Here we use the notation of [5]. The reader is encouraged to take a look there for notational purposes.

Take $c := \cos(\pi/4) = \sqrt{2}/2$ and $0 \leq \varepsilon < 1$. Define the matrices

$$X_p := \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad U_p := \begin{pmatrix} c & 0 \\ 0 & c \\ c & 0 \\ 0 & c \end{pmatrix}, \quad M := U_p \begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix}.$$

These matrices satisfy the conditions posed in [5]:

- Principal alignment: $X_p^T U_p = \begin{pmatrix} c & 0 \\ 0 & c \end{pmatrix}$.
- Principal angles between X_p and U_p are in $[0, \pi/4]$.
- $U = U_p$ since $Q = I$.

Now consider the following tangent vector of unit Frobenius norm:

$$\Delta = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

It is clearly a tangent vector of $[X_p]$ since $X_p^T\Delta = 0$. The Hessian of f_{full} at $[X_p]$ in the direction of Δ satisfies (see equation (4.2) in [5])

$$\mathrm{Hess} f_{full}([X_p])[\Delta,\Delta] = -2\,\mathrm{Tr}(M^T\Delta\Delta^T(I - X_pX_p^T)M) + \|(\Delta X_p^T + X_p\Delta^T)M\|_F^2.$$

Simple calculation shows that

$$\mathrm{Hess} f_{full}([X_p])[\Delta,\Delta] = -2c^2 + (1+\varepsilon^2)c^2.$$

Hence for $\varepsilon < 1$, we have $\mathrm{Hess} f_{full}([X_p])[\Delta,\Delta] < 0$ and the f_{full} is non-convex which is in contrast with Corollary 4.

Instead, our Theorem 2.21 guarantees convexity when φ depends on the spectral gap. Since f is smooth, the function is geodesically convex on $N_*(\varphi)$ if and only if its Riemannian Hessian is positive definite on $N_*(\varphi)$; see, e.g., [21, Thm. 11.23]. We will therefore compute the eigenvalues of $\mathrm{Hess}\, f$ based on its matrix representation. This requires us to first vectorize the tangent space.

From (1.3.1.3), a matrix G is a tangent vector if and only if $G^TX = 0$. Hence, taking $X_\perp \in \mathbb{R}^{n\times(n-k)}$ orthonormal such that $\mathcal{X}^\perp = \mathrm{Span}(X_\perp)$, we have the equivalent definition

$$T_X\,\mathrm{Gr}(n,k) = \{X_\perp M \colon M \in \mathbb{R}^{(n-k)\times k}\}.$$

The matrix M above can be seen as the coordinates of $G = X_\perp M$ in the basis $X_\perp$. More specifically, by using the linear isomorphism $\mathrm{vec}\colon \mathbb{R}^{n\times k} \to \mathbb{R}^{nk}$ that stacks all columns of a matrix under each other, we can define the tangent vectors of $\mathrm{Gr}(n,k)$ as standard (column) vectors in the following way:

$$\mathrm{vec}(G) = \mathrm{vec}(X_\perp M) = (I_k \otimes X_\perp)\,\mathrm{vec}(M).$$

Here, the Kronecker product $\otimes$ appears due to [48, Lemma 4.3.1]. By well-known properties of $\otimes$ (see, e.g., [48, Chap. 4.2]), the matrix $I_k \otimes X_\perp$ has orthonormal columns. We have thus obtained an orthonormal basis for the (vectorized) tangent space. With this setup, we can now construct the Hessian.

Lemma 2.19 *Let $I_k \otimes X_\perp$ be the orthonormal basis for the vectorization of $T_{\mathcal{X}}\,\mathrm{Gr}(n,k)$. Then the Riemannian Hessian of f at $\mathcal{X}$ in that basis has the symmetric matrix representation*

$$H_X = 2(X^TAX \otimes I_{n-k} - I_k \otimes X_\perp^TAX_\perp). \tag{2.5.0.2}$$

Furthermore, with $1 \le i \le k$ and $1 \le j \le n-k$ its $k(n-k)$ eigenvalues satisfy

$$\lambda_{i,j}(H_X) = 2(\lambda_i(X^TAX) - \lambda_j(X_\perp^TAX_\perp)).$$

Proof Since vec is a linear isomorphism, the symmetric matrix H_X satisfies

$$\mathrm{Hess}\, f(X)[X_\perp M, X_\perp M] = \langle \mathrm{vec}(M), H_X\,\mathrm{vec}(M)\rangle, \qquad \forall M \in \mathbb{R}^{n\times(n-k)},$$

where $\langle \cdot, \cdot \rangle$ is the Euclidean inner product. Define $m = \mathrm{vec}(M)$. Plugging in the formula (2.1.0.1) for $\mathrm{Hess}\, f$, we calculate

$$\begin{aligned}
\mathrm{Hess}\, f(X)[X_\perp M, X_\perp M] &= 2\langle X_\perp M, X_\perp M X^T A X - A X_\perp M \rangle \\
&= 2\langle (I \otimes X_\perp)m, (X^T A X \otimes X_\perp)m - (I \otimes A X_\perp)m \rangle \\
&= 2\langle m, (I \otimes X_\perp)^T (X^T A X \otimes X_\perp - I \otimes A X_\perp)m \rangle \\
&= 2\langle m, (X^T A X \otimes I - I \otimes X_\perp^T A X_\perp)m \rangle
\end{aligned}$$

Here, we used typical calculus rules for the Kronecker product (see, e.g., [48, Chap. 4.2]). We recognize the matrix H_X directly.

The eigenvalues of (2.5.0.2) can be directly obtained using [48, Thm. 4.4.5]. ■

Taking $X = V_\alpha$ and $X_\perp = V_\beta$, Lemma 2.19 shows immediately that the minimal eigenvalue of $\mathrm{Hess}\, f(\mathcal{V}_\alpha)$ is equal to $2\delta = 2(\lambda_k - \lambda_{k+1})$. Since $\delta > 0$, $\mathrm{Hess}\, f$ will remain strictly positive definite in a neighbourhood of $\mathcal{V}_\alpha$ by continuity. To quantify this neighbourhood, we will connect $\mathcal{V}_\alpha$ to an arbitrary $\mathcal{X}$ using a geodesic and see how this influences the bounds of Lemma 2.19. This also requires connecting $\mathcal{V}_\beta$ to $\mathcal{X}^\perp$. The next lemma shows that both geodesics are closely related. Recall that $\sin(t\theta)$ and $\cos(t\theta)$ denote diagonal matrices of size $k \times k$. For convenience, we will denote by O a zero matrix whose dimensions are clear from the context and is not always square.

Lemma 2.20 *Let $X, Y \in \mathbb{R}^{n\times k}$ be such that $X^T X = Y^T Y = I_k$ with $k \leq n/2$. Denote the principal angles between $\mathrm{Span}(X)$ and $\mathrm{Span}(Y)$ by $\theta_1 \leq \cdots \leq \theta_k$ and assume that $\theta_k < \pi/2$. Choose $X_\perp, Y_\perp \in \mathbb{R}^{n\times(n-k)}$ such that $X_\perp^T X_\perp = Y_\perp^T Y_\perp = I_{n-k}$ and $\mathrm{Span}(X_\perp) = \mathrm{Span}(X)^\perp$, $\mathrm{Span}(Y_\perp) = \mathrm{Span}(Y)^\perp$. Define the curves*

$$\begin{aligned}
\gamma(t)\colon [0,1] \to \mathbb{R}^{n\times k}, &\qquad t \mapsto X V_1 \cos(t\theta) + X_\perp V_2 \begin{bmatrix} O \\ \sin(t\theta) \end{bmatrix}, \\
\gamma_\perp(t)\colon [0,1] \to \mathbb{R}^{n\times(n-k)}, &\qquad t \mapsto X_\perp V_2 \begin{bmatrix} I & \\ & \cos(t\theta) \end{bmatrix} - X V_1 \begin{bmatrix} O & \sin(t\theta) \end{bmatrix},
\end{aligned}$$

where the orthogonal matrices V_1, V_2 are the same as in Lemma 2.5. Then $\mathrm{Span}(\gamma(t))$ is the connecting geodesic on $\mathrm{Gr}(n,k)$ from $\mathrm{Span}(X)$ to $\mathrm{Span}(Y)$. Likewise, $\mathrm{Span}(\gamma_\perp(t))$ is a connecting geodesic on $\mathrm{Gr}(n, n-k)$ from $\mathrm{Span}(X_\perp)$ to $\mathrm{Span}(Y_\perp)$. Furthermore, $\gamma(t)$ and $\gamma_\perp(t)$ are orthonormal matrices for all t.

Proof Assume $\theta_1 = \cdots = \theta_r = 0$, where $r = 0$ means that $\theta_1 > 0$. Like in the proof of Prop. 2.6, the CS decomposition of X and Y from Lemma 2.5 can be written in terms of their principal angles $\theta_1, \ldots, \theta_k$. Since $\theta_k < \pi/2$ and

$n \leq k/2$, this gives after dividing certain block matrices the relations

$$Y^T X = U_1 \cos(\theta) V_1^T, \qquad Y^T X_\perp = U_1 \begin{bmatrix} O_{k\times(n-2k)} & \sin(\theta) \end{bmatrix} V_2^T$$
$$Y_\perp^T X = U_2 \begin{bmatrix} O_{(n-2k)\times k} \\ \sin(\theta) \end{bmatrix} V_1^T, \qquad Y_\perp^T X_\perp = U_2 \begin{bmatrix} -I_{n-2k} & \\ & -\cos(\theta) \end{bmatrix} V_2^T,$$

where U_1, V_1 and U_2, V_2 are orthogonal matrices of size $k\times k$ and $(n-k)\times(n-k)$, resp.

Denote $\mathcal{X} = \mathrm{Span}(X)$ and $\mathcal{Y} = \mathrm{Span}(Y)$. By definition, the connecting geodesic $\gamma(t)$ is determined by the tangent vector $\mathrm{Log}_{\mathcal{X}}(\mathcal{Y})$, which can be computed from (1.3.1.6). To this end, we first need the compact SVD of $M := X_\perp X_\perp^T Y (X^T Y)^{-1}$. Substituting the results from above, we get (cfr. (2.2.2.5))

$$M = X_\perp V_2 \begin{bmatrix} O_{(n-2k)\times k} \\ \sin(\theta) \end{bmatrix} U_1^T U_1 \left(\cos(\theta)\right)^{-1} V_1^T = X_\perp V_2 \begin{bmatrix} O_{(n-2k)\times k} \\ I_k \end{bmatrix} \tan(\theta) V_1^T.$$

Observe that this is a compact SVD. Applying (1.3.1.6), we therefore get

$$G := \mathrm{Log}_{\mathcal{X}}(\mathcal{Y}) = U\Sigma V^T \quad \text{with } U = X_\perp V_2 \begin{bmatrix} O \\ I_k \end{bmatrix},\ \Sigma = \theta,\ V = V_1$$

and from (1.3.1.4), the connecting geodesic satisfies

$$\mathrm{Exp}_{\mathcal{X}}(tG) = \mathrm{Span}(\, XV_1 \cos(t\theta) + X_\perp V_2 \begin{bmatrix} O \\ I_k \end{bmatrix} \sin(t\theta)\,).$$

We have proven the stated formula for $\gamma(t)$. Verifying that $\gamma(t)^T\gamma(t) = I_k$ follows from a simple calculation that uses $\cos^2(t\theta) + \sin^2(t\theta) = I_k$.

Denote $\mathcal{X}^\perp = \mathrm{Span}(X_\perp)$ and $\mathcal{Y}^\perp = \mathrm{Span}(Y_\perp)$. To prove $\gamma_\perp(t)$, we proceed similarly by computing $G^\perp := \mathrm{Log}_{\mathcal{X}^\perp}(\mathcal{Y}^\perp)$, which requires now the SVD of $M^\perp := XX^T Y_\perp (X_\perp^T Y_\perp)^{-1}$. Again substituting the results from the CS decomposition, we get

$$\begin{aligned} M^\perp &= XV_1 \begin{bmatrix} O_{k\times(n-2k)} & \sin(\theta) \end{bmatrix} U_2^T U_2 \begin{bmatrix} -I_{n-2k} & \\ & -\cos(\theta) \end{bmatrix}^{-1} V_2^T \\ &= XV_1 \begin{bmatrix} O_{k\times(n-2k)} & -\tan(\theta) \end{bmatrix} V_2^T \end{aligned}$$

Since (1.3.1.6) requires a compact SVD with a *square* Σ, we rewrite this as

$$M^\perp = \begin{bmatrix} \widetilde{X} & XV_1 \end{bmatrix} \begin{bmatrix} O_{(n-2k)\times(n-2k)} & \\ & -\tan(\theta) \end{bmatrix} V_2^T$$

where $\widetilde{X}$ contains $n-2k$ columns that are orthonormal to X (the final result will not depend on $\widetilde{X}$). Let $\theta_1^\perp \leq \cdots \leq \theta_{n-k}^\perp$ denote the principal angles

between $\mathcal{X}^\perp$ and $\mathcal{Y}^\perp$. Up to zero angles, they are the same as those between $\mathcal{X}$ and $\mathcal{Y}$. Since $k \leq n/2$, we thus have

$$\theta_1^\perp = \cdots = \theta_{n-2k}^\perp = 0,\ \theta_{n-2k+1}^\perp = \theta_1, \ldots, \theta_{n-k}^\perp = \theta_k.$$

Applying (1.3.1.6) with these principal angles, we obtain

$$G^\perp := \mathrm{Log}_{\mathcal{X}^\perp}(\mathcal{Y}^\perp) = U\Sigma V^T \quad \text{with } U = -\begin{bmatrix}\widetilde{X} & XV_1\end{bmatrix},\ \Sigma = \theta^\perp,\ V = V_2.$$

From (1.3.1.4), the corresponding geodesic satisfies

$$\begin{aligned}\mathrm{Exp}_{\mathcal{X}^\perp}(tG^\perp) &= \mathrm{Span}(\, X_\perp V_2 \cos(t\theta^\perp) - \begin{bmatrix}\widetilde{X} & XV_1\end{bmatrix} \sin(t\theta^\perp)\,) \\ &= \mathrm{Span}(\, X_\perp V_2 \begin{bmatrix} I_{n-2k} & \\ & \cos(t\theta)\end{bmatrix} - \begin{bmatrix} O_{n\times(n-2k)} & XV_1 \sin(t\theta)\end{bmatrix}\,).\end{aligned}$$

Rewriting the block matrix, we have proven $\gamma_\perp(t)$. Its orthonormality is again a straightforward verification. ∎

With the previous lemma, we can now investigate the Riemannian Hessian of f near $\mathcal{V}_\alpha$ when it is given in the matrix form H_X of Lemma 2.19. Let $\mathcal{X} = \mathrm{Span}(X) \in \mathrm{Gr}(n,k)$ with orthonormal X. Its principal angles with $\mathcal{V}_\alpha$ are $\theta_1 \leq \cdots \leq \theta_k < \pi/2$. Use the substitutions $X \mapsto V_\alpha, Y \mapsto X$ and $X_\perp \mapsto V_\beta, Y_\perp \mapsto X_\perp$ in Lemma 2.20 to define the geodesics $\gamma(t)$ and $\gamma_\perp(t)$ that connect $\mathcal{V}_\alpha$ to $\mathcal{X}$, and $\mathcal{V}_\beta$ to $\mathcal{X}^\perp$, resp. Denoting

$$C := \cos(\theta),\ S := \sin(\theta),\ \widetilde{C} := \begin{bmatrix} I & \\ & C\end{bmatrix},\ \widetilde{S} := \begin{bmatrix} O \\ S\end{bmatrix},$$

we get the following expressions for the geodesics:

$$\gamma(t) = V_\alpha V_1 C + V_\beta V_2 \widetilde{S}, \quad \gamma_\perp(t) = V_\beta V_2 \widetilde{C} - V_\alpha V_1 \widetilde{S}^T.$$

Recall that H_X is defined using $X^T A X$ and $X_\perp^T A X_\perp$. Since $\gamma(1) = XQ_1$ and $\gamma_\perp(1) = X^\perp Q_2$ for some orthogonal matrices Q_1, Q_2, we can write with $A = V_\alpha \Lambda_\alpha V_\alpha^T + V_\beta \Lambda_\beta V_\beta^T$ that

$$\begin{aligned} Q_1^T X^T A X Q_1 &= \gamma(1)^T A \gamma(1) \\ &= C\,(V_1^T \Lambda_\alpha V_1)\, C + \widetilde{S}^T\,(V_2^T \Lambda_\beta V_2)\,\widetilde{S} \\ Q_2^T X_\perp^T A X_\perp Q_2 &= \gamma_\perp(1)^T A \gamma_\perp(1) \\ &= \widetilde{C}\,(V_2^T \Lambda_\beta V_2)\,\widetilde{C} + \widetilde{S}\,(V_1^T \Lambda_\alpha V_1)\,\widetilde{S}^T. \end{aligned} \tag{2.5.0.3}$$

Here we used simplifications like $V_\beta^T A V_\alpha = V_\beta^T V_\alpha \Lambda_\alpha = 0$.

A simple bounding of the eigenvalues of the difference of these matrices results in the main result.

Theorem 2.21 *Let $k \leq n/2$. Define the neighbourhood*

$$B_* = \left\{ \mathcal{X} \in \mathrm{Gr}(n,k) \colon \sin^2(\theta_k(\mathcal{X}, \mathcal{V}_\alpha)) \leq \frac{\delta}{\lambda_1 + \lambda_k} \right\},$$

then f is geodesically convex on B_*.

Proof Our aim is to show that $\lambda_{i,j}(H_X)$ remains positive given the bound on θ_k. From Lemma 2.19, we see that

$$\lambda_{\min}(H_X) \geq 0 \quad \iff \quad \lambda_{\min}(X^T A X) \geq \lambda_{\max}(X_\perp^T A X_\perp). \tag{2.5.0.4}$$

Since Q_1, Q_2 are orthogonal in (2.5.0.3), it suffices to find a lower and upper bound of, resp.,

$$\begin{aligned}
\lambda_{\min}(X^T A X) &= \lambda_{\min}(C\,(V_1^T \Lambda_\alpha V_1)\,C + \widetilde{S}^T\,(V_2^T \Lambda_\beta V_2)\,\widetilde{S}) \\
\lambda_{\max}(X_\perp^T A X_\perp) &= \lambda_{\max}(\widetilde{C}\,(V_2^T \Lambda_\beta V_2)\,\widetilde{C} + \widetilde{S}\,(V_1^T \Lambda_\alpha V_1)\,\widetilde{S}^T).
\end{aligned}$$

Standard eigenvalue inequalities for symmetric matrices (see, e.g., [47, Cor. 4.3.15]) give

$$\begin{aligned}
\lambda_{\min}(X^T A X) &\geq \lambda_{\min}(C\,(V_1^T \Lambda_\alpha V_1)\,C) + \lambda_{\min}(\widetilde{S}^T\,(V_2^T \Lambda_\beta V_2)\,\widetilde{S}) \\
\lambda_{\max}(X_\perp^T A X_\perp) &\leq \lambda_{\max}(\widetilde{C}\,(V_2^T \Lambda_\beta V_2)\,\widetilde{C}) + \lambda_{\max}(\widetilde{S}\,(V_1^T \Lambda_\alpha V_1)\,\widetilde{S}^T).
\end{aligned}$$

Recall that $\lambda_1 \geq \cdots \geq \lambda_n$ are the eigenvalues of A. Since $\widetilde{S}$ is a tall rectangular matrix, we apply the generalized version of Ostrowski's theorem from [46, Thm. 3.2] to each term above[7] and obtain

$$\begin{aligned}
\lambda_{\min}(C\,(V_1^T \Lambda_\alpha V_1)\,C) &\geq \lambda_{\min}(C^2)\lambda_{\min}(\Lambda_\alpha) = \cos^2(\theta_k)\lambda_k \\
\lambda_{\min}(\widetilde{S}^T\,(V_2^T \Lambda_\beta V_2)\,\widetilde{S}) &\geq \lambda_{\min}(\tilde{S}^T \tilde{S})\lambda_{\min}(\Lambda_\beta) = \sin^2(\theta_1)\lambda_n,
\end{aligned}$$

since the matrices V_1, V_2 are orthogonal and $\theta_1 \leq \cdots \leq \theta_k < \pi/2$. Adding this gives the lower bound

$$\lambda_{\min}(X^T A X) \geq \cos^2(\theta_k)\lambda_k + \sin^2(\theta_1)\lambda_n \geq \cos^2(\theta_k)\lambda_k. \tag{2.5.0.5}$$

Likewise, using the block structure of $\widetilde{S}$, we get

$$\begin{aligned}
\lambda_{\max}(\widetilde{C}\,(V_2^T \Lambda_\beta V_2)\,\widetilde{C}) &\leq \lambda_{\max}(C^2)\lambda_{\max}(\Lambda_\beta) = \cos^2(\theta_1)\lambda_{k+1} \\
\lambda_{\max}(\widetilde{S}\,(V_1^T \Lambda_\alpha V_1)\,\widetilde{S}^T) &= \lambda_{\max}(S\,(V_1^T \Lambda_\alpha V_1)\,S) \\
&\leq \lambda_{\max}(S^2)\lambda_{\max}(\Lambda_\alpha) = \sin^2(\theta_k)\lambda_1
\end{aligned}$$

and thus

$$\lambda_{\max}(X_\perp^T A X_\perp) \leq \cos^2(\theta_1)\lambda_{k+1} + \sin^2(\theta_k)\lambda_1 \leq \lambda_{k+1} + \sin^2(\theta_k)\lambda_1. \tag{2.5.0.6}$$

[7] Observe that the cited theorem orders the eigenvalues inversely to the convention used in this paper.

The condition (2.5.0.4) is thus satisfied when

$$\cos^2(\theta_k)\lambda_k = \lambda_k - \sin^2(\theta_k)\lambda_k \geq \lambda_{k+1} + \sin^2(\theta_k)\lambda_1,$$

which reduces to the bound on θ_k in the statement of the theorem.

It remains to show that B_* is an open totally geodesically convex set. Since $\lambda_1 \geq \lambda_k \geq \lambda_{k+1} \geq 0$, we get

$$\frac{\lambda_k - \lambda_{k+1}}{\lambda_1 + \lambda_k} \leq \frac{\lambda_k}{2\lambda_k} = \frac{1}{2}.$$

Hence, $B_* = N_*(\varphi)$ with $\varphi \leq \pi/4$ since $\sin^2(\pi/4) = 1/2$. ■

If $k = 1$, the proof above can be simplified.

Corollary 2.22 *Let $k = 1$ and define the neighbourhood*

$$B_* = \left\{ \mathcal{X} \in \mathrm{Gr}(n,1) \colon \sin^2(\theta_1(\mathcal{X}, \mathcal{V}_\alpha)) \leq \frac{\delta}{\delta + \lambda_1 - \lambda_n} \right\}.$$

Then f is geodesically convex on B_*.

Proof Since $k = 1$, there is no need to simplify the bounds (2.5.0.5) and (2.5.0.6) as was done above. This gives that f is convex as long as

$$\cos^2(\theta_1)\lambda_1 + \sin^2(\theta_1)\lambda_n \geq \cos^2(\theta_1)\lambda_2 + \sin^2(\theta_1)\lambda_1.$$

Rewriting leads directly to the stated condition on $\sin^2(\theta_1)$. ■

Remark that optimizing f on $\mathrm{Gr}(n,1)$ is equivalent to

$$\min_{x \in \mathbb{R}^n} -x^T A x \qquad \text{s.t.} \quad \|x\| = 1, \tag{2.5.0.7}$$

which is the minimization of the Rayleigh quotient problem on the unit sphere $S^{n-1} = \{x \in \mathbb{R}^n \colon x^T x = 1\}$. Cor. 2.22 can therefore also be phrased in terms of a geodesically convex region for this problem. Denoting a unit norm top eigenvector of A by v_1 and using that $\sin^2\theta_1 = 1 - \cos^2\theta_1$, we get that (2.5.0.7) is geodesically convex on

$$\hat{B}_* = \left\{ x \in S^{n-1} \colon (x^T v_1)^2 \geq 1 - \frac{\delta}{\delta + \lambda_1 - \lambda_n} \right\}.$$

This result can now be directly compared to [50, Lemma 7] where the corresponding region is defined as $(x^T v_1)^2 \geq 1 - \frac{\delta}{\delta + \lambda_1}$. This is a stricter condition and our result is therefore a small improvement.

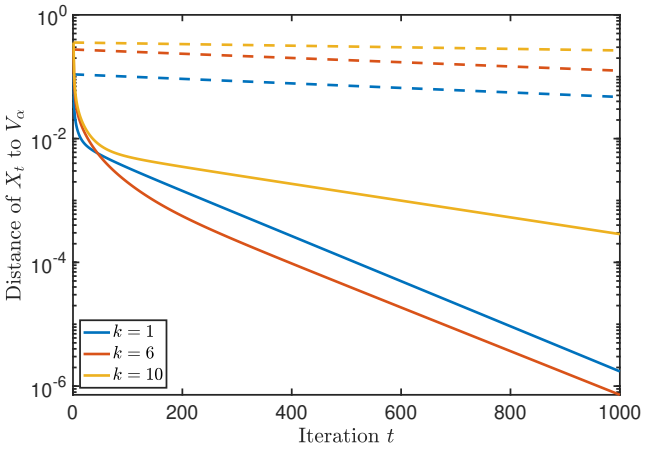


Figure 2.1: gradient descent along geodesics for the block Rayleigh quotient of size k applied to a discretized 3D Laplacian matrix. The full lines correspond to the experimental values and the dashed lines to the theoretical upper bounds.

2.6 Numerical experiment

We report on a small numerical experiment to verify the convergence rates proven above. The gradient descent iteration with fixed step size was implemented in MATLAB using the geodesic formula (1.3.1.4).

As first test matrix, we took the standard 3D Laplacian on a unit cube, discretized with finite differences and zero Dirichlet boundary conditions. The size of the matrix A is $n = 400$. We tested a few values for the block size k. They are depicted in the table below, together with other parameters that are relevant for Theorem 2.10.

k	δ	$\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$
1	$0.0665\ldots$	$0.113\ldots$
6	$0.0665\ldots$	$0.280\ldots$
10	$0.0262\ldots$	$0.350\ldots$

In Figure 2.1, the convergence of the Riemannian distance is visible in addition to the theoretical convergence rate of Theorem 2.10. We see that in all cases, these bounds on the convergence are valid (in particular, exponential) although they are rather conservative.

For completeness, we implemented gradient descent starting from a subspace $\mathcal{X}_0$ far away from the optimum. In that case, Theorem 2.10 does not apply since, if $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) > \frac{\pi}{2}$, the step size $\eta \leq \cos(\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha))/L$ is or will become eventually negative. However, a meaningful choice for η is given by Proposition 2.16 of Section 2.4, where we prove a local linear convergence rate for the function values of the iterates for step size $\eta = 1/L$.

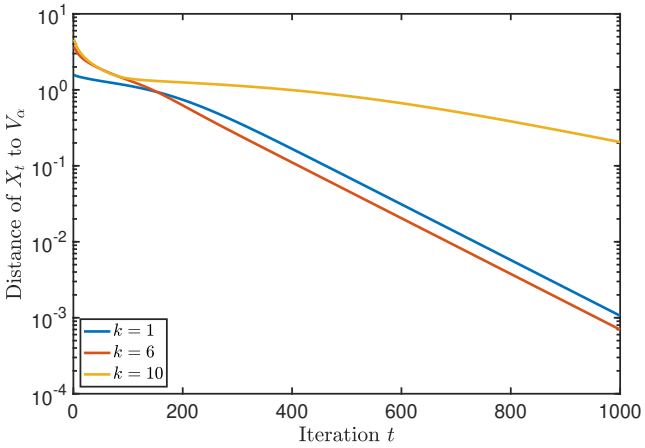


Figure 2.2: Same matrix from Figure 2.1 but such that $\text{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \gg \pi/2$ and with fixed step size $1/L$.

We see in Figure 2.2 that despite the seemingly bad initial guess, gradient descent converges globally with a linear rate.

In the second test, we investigate the convergence when the spectral gap δ is small or zero. In particular, we take $A = VDV^T \in \mathbb{R}^{1000\times 1000}$ with V a random orthogonal matrix and D contains the eigenvalues

$$\lambda_1 = 3,\ \lambda_2 = 2,\ \lambda_3 = 1 + 10^{-2} + 10^{-6},\ \lambda_4 = 1 + 10^{-6},\ \lambda_5 = \lambda_6 = 1.$$

The other eigenvalues are equidistantly distributed between 0.1 and 0.2. The block size and other relevant parameters for the test are described below. Since the convergence for small δ slows down considerably after the first 5 iterations, we apply the bounds of Theorem 2.12 at iteration $t = 6$ (and treat this as the start with $t = 0$).

k	δ	$\text{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$	$\text{dist}(\mathcal{X}_6, \mathcal{V}_\alpha)$
2	$0.99\ldots$	$0.051\ldots$	$0.001\ldots$
3	10^{-2}	$0.055\ldots$	$0.031\ldots$
4	10^{-6}	$0.063\ldots$	$0.045\ldots$
5	0	$0.070\ldots$	$0.054\ldots$

The convergence in function value is visible in Figure 2.3. Observe that we have displayed a logarithmic scale for both axes whereas before the figure had a logarithmic scale only for y-axis. Algebraic convergence like $1/t$ is therefore visible as a straight line. We see in the figure that the convergence is not easily described, and that there is no clear difference between zero or small gap. However, the upper bounds of Theorem 2.12 are again valid. In addition, when the gap is not small, the convergence is clearly faster.

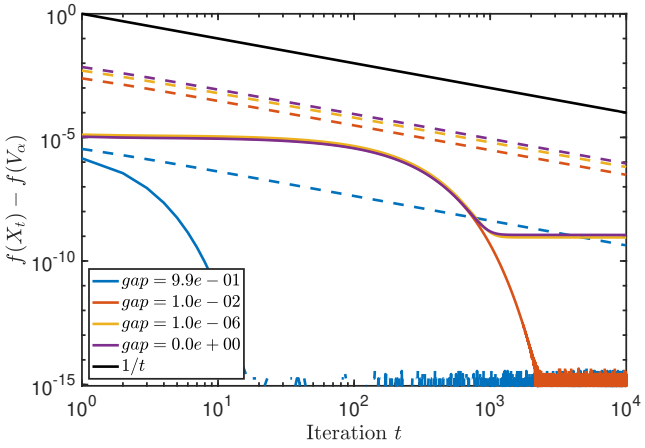


Figure 2.3: Gradient descent along geodesics for the block Rayleigh quotient of size k applied to a random matrix with small spectral gaps. The full lines correspond to the experimental values and the dashed lines to the theoretical upper bounds of Theorem 2.12. Each color corresponds to a certain spectral gap δ.

As before, we test the behaviour of gradient descent starting from an initial guess far away from the optimum. We use again step size $1/L$; see Theorem 2.18. In Figure 2.4 we show the convergence of gradient descent for the problem defined by matrix A with this step size.

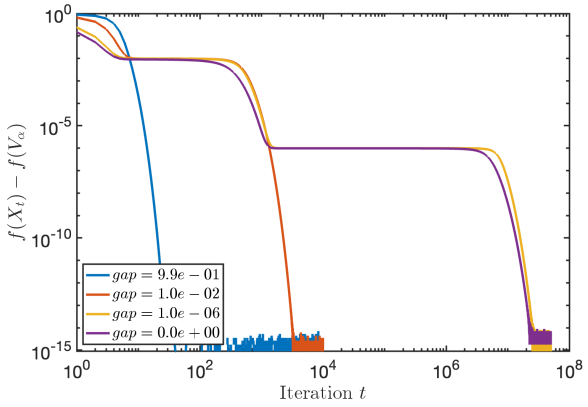


Figure 2.4: Same matrices with small spectral gap from Figure 2.3 but such that $\text{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \gg \pi/2$ and with fixed step size $1/L$.

We observe again that the local nature of our theoretical results is quite pessimistic: the algorithm converges with an algebraic rate even with a bad initial guess but it shows eventually linear convergence.

3 Distributed principal component analysis with limited communication

We now discuss the first application of the theory presented in Section 2. This has to do with principal component analysis in a data-parallel regime, where the different agents (each one of them holds some batch of the data) communicate in a low bit precision. We follow the exposition of our work [8], which actually came before our work [14] and gave motivation for the development of the general theory presented in Section 2. In the context of this thesis however, we believe it is better to present it as a consequence of this theory.

3.1 Introduction

Something important to notice is that [8] deals only with the computation of the leading principal component, i.e. only the leading eigenvector of a covariance matrix. A suitable space to formulate this problem as an optimization problem is the sphere. Notice that the sphere is not the same with the manifold $\mathrm{Gr}(n, 1)$. Actually, $\mathrm{Gr}(n, 1)$ is a sphere but with the two hemishperes merged and recording only the *direction* of some vector. This is still a manifold and is called projective space. From a mathematical point of view though, working in the sphere or in the projective space is essentially the same.

Using the theory of Section 2, we could comfortably extend the results of [8] in the block case. However, we do not believe that this adds substantially to the scientific value of the exposed ideas and we shall stick to the case $k = 1$. For completeness, we will re-prove the convexity-like properties developed in Section 2 for the $k = 1$ case in the sphere.

To the best of our knowledge, the work presented in this section is the first one to focus on the *bandwidth cost* of distributed PCA, i.e. the number of bits which need to be transmitted for achieving computation of the first principal component up to some accuracy. This is a significant bottleneck in distributed systems and many works have dealt with it for other classic problems, e.g. [49, 52, 108]. On the other hand, many works have dealt with the problem of distributed PCA focusing only on *latency cost*, i.e. the number of required communication rounds.

Our main contribution is a new algorithm for distributed leading eigenvector computation, which specifically minimizes the total number of bits sent and received by the computing nodes. To that end, we use a standard quantization scheme developed in [31]. The theoretical analysis is done using special cases of properties developed in Section 2.

3.2 Setting and Related Work

Setting. We assume to be given m total samples coming from some distribution $\mathcal{D}$, organized as a global $m \times n$ data matrix M, which is partitioned row-wise among p processors, with node i being assigned the matrix M_i, consisted by m_i consecutive rows, such that $\sum_{i=1}^{p} m_i = m$. As is common, let $A := M^T M = \sum_{i=1}^{p} M_i^T M_i$ be the global covariance matrix, and $A_i := M_i^T M_i$ the local covariance matrix owned by the node i. We denote by $\lambda_1, \lambda_2, ..., \lambda_n$ the eigenvalues of A in descending order and by $v_1, v_2, ..., v_n$ the corresponding eigenvectors. We can approximate the leading eigenvector by solving the following empirical risk minimization problem up to accuracy ε:

$$x^\star = \text{argmin}_{x \in \mathbb{R}^n \setminus \{0\}} \left(-\frac{x^T A x}{\|x\|^2} \right) = \text{argmin}_{x \in \mathbb{S}^{n-1}} \left(-x^T A x \right), \qquad (3.2.0.1)$$

where $\mathbb{S}^{n-1}$ is the $(n-1)$-dimensional sphere. When the spectral gap $\delta := \lambda_1 - \lambda_2$ is strictly positive, the optimum x^* is unique up to a change of sign.

We define $f : \mathbb{S}^{n-1} \to \mathbb{R}$, with $f(x) = -x^T A x$ and $f_i : \mathbb{S}^{n-1} \to \mathbb{R}$, with $f_i(x) = -x^T A_i x$. Since the inner product is bilinear, we can write the global cost as the sum of the local costs:

$$f(x) = \sum_{i=1}^{p} f_i(x).$$

Related Work. Lately, there has been a significant amount of research on efficient variants of PCA and related problems [19, 90, 99, 100, 117, 118]. In order to keep this discussion as simple as possible, we focus on related work on communication-efficient algorithms. In particular, we discuss the relationship to previous *round-efficient* algorithms; to our knowledge, what presented in this section is the first work to specifically focus on the bit complexity of this problem in the setting where data is randomly partitioned. More precisely, previous work on this variant implicitly assumes that algorithms are able to transmit real numbers at unit cost.

The straightforward approach to solve the minimization problem (3.2.0.1) in a distributed setting, where the data rows are partitioned, would be to use a distributed version of the power method, Riemannian gradient descent (RGD), or the Lanczos algorithm. In order to achieve an ε-approximation of the minimizer $x^\star$, the latter two algorithms require $\tilde{\mathcal{O}}\left(\frac{\lambda_1}{\delta} \log(1/\varepsilon)\right)$ rounds, where the $\tilde{\mathcal{O}}$ notation hides poly-logarithmic factors in n. Distributed Lanczos and accelerated RGD would improve this by an $\mathcal{O}(\sqrt{\lambda_1/\delta})$ factor. However, Garber et al. [38] point out that, when δ is small, e.g. $\delta = \Theta(1/\sqrt{Kp})$, then unfortunately the number of communication rounds would increase with the sample size, which renders these algorithms non-scalable.

Standard distributed convex approaches, e.g. [53, 101], do not directly extend to our setting due to non-convexity and the unit-norm constraint. Garber et al. [38] proposed a variant of the Power Method, called *Distributed Shift-and-Invert (DSI)*, which converges in roughly $\mathcal{O}\left(\sqrt{\frac{b}{\delta\sqrt{p}}}\,\mathrm{Log}^2(1/\varepsilon)\log(1/\delta)\right)$ rounds, where b is a bound on the squared ℓ_2-norm of the data. Huang and Pan [50] aimed to improve the dependency of the algorithm on ε and δ, and proposed an algorithm called *Communication-Efficient Distributed Riemannian Eigensolver (CEDRE)*. This algorithm is shown to have round complexity $\mathcal{O}\left(\frac{b}{\delta\sqrt{p}}\log(1/\varepsilon)\right)$, which does not scale with the sample size for $\delta = \Omega(1/\sqrt{Kp})$, and has only logarithmic dependency on the accuracy ε.

Technical issues in [50]. Huang and Pan [50] proposed an interesting approach, which could provide the most round-efficient distributed algorithm to date. Despite the fact that we find the main idea of this paper very creative, we have unfortunately identified a significant gap in their analysis, which we now outline.

Specifically, one of their main results, Theorem 3, uses the local PL condition shown in [117]; yet, the application of this result is invalid, as it is done without knowing in advance that the iterates of the algorithms continue to remain in the ball of initialization. This is compounded by another error on the constant of the used PL condition (Lemma 2), which we believe is caused by a typo in the last part of the proof of Theorem 4 in [117]. This typo is unfortunately propagated into their proof. More precisely, the objective is indeed gradient dominated, but with a constant which vanishes when we approach the equator, in contrast with the $2/\delta$ which is claimed *globally* (we reprove this result independently in Proposition 2.8). Thus, starting from a point lying in some ball of the minimizer can lead to a new point where the objective satisfies a PL condition, but with a worse constant.

This is a non-trivial technical problem which, in the case of gradient descent, can be addressed by a choice of the learning rate depending on the initialization. Given these issues, we perform a new and formally-correct analysis for gradient descent based on the convexity-like properties derived in Section 2, which guarantee convergence directly in terms of the distance of iterates to the optimum, and not just function values. We would like however to note that our focus in this section is on bit and not round complexity.

3.3 Computing the Leading Eigenvector in One Node

Here, we essentially make a recap of the theory developed in Section 2 for $k = 1$, using mostly the terminology of [8]. This has some value as now the manifold of interest is the sphere. See also our introduction on the geometry of the sphere in Section 1.3.1.1.

3.3.1 Convexity-like Properties and Smoothness

Our problem reads as

$$\min_{x\in\mathbb{S}^{n-1}} -x^TAx$$

where $A = M^TM$ is the global covariance matrix. If $\delta = \lambda_1 - \lambda_2 > 0$, this problem has exactly two global minima: v_1 and $-v_1$. Let $x \in \mathbb{S}^{n-1}$ be an arbitrary point. Then x can be written in the form $x = \sum_{i=1}^n \alpha_i v_i$. Fixing the minimizer v_1, we have that a ball in $\mathbb{S}^{n-1}$ around v_1 is of the form

$$B_a = \{x \in \mathbb{S}^{d-1} \mid \alpha_1 \geq a\} = \{x \in \mathbb{S}^{d-1} \mid \langle x, v_1\rangle \geq a\} \tag{3.3.1.1}$$

for some a. Without loss of generality, we may assume $a > 0$ (otherwise, consider the ball around $-v_1$ to establish convergence to $-v_1$).

We investigate the convexity properties of the function $-x^TAx$. In particular, we prove that this function is weakly-quasi convex with constant $2a$ in the ball B_a (that is to say $(2a, 0) - WQSC$, see Definition 1.22).

Proposition 3.1 *The function $f(x) = -x^TAx$ satisfies*

$$2a(f(x) - f^*) \leq \langle \mathrm{grad} f(x), -\mathrm{Log}_x(x^*)\rangle$$

for any $x \in B_a$ with $a > 0$.

Proof For any $x \in B_a$, we can write

$$x = \sum_{i=1}^n \alpha_i v_i, \qquad Ax = \sum_{i=1}^d \lambda_i\alpha_i v_i \tag{3.3.1.2}$$

for some scalars α_i. Recall that $\alpha_1 \geq a > 0$ by (3.3.1.1).

With the orthogonal projector $P_x = I - xx^T$ onto the tangent space $T_x\mathbb{S}^{n-1}$, we get that

$$\begin{aligned}\langle \mathrm{grad} f(x), -\mathrm{Log}_x(x^*)\rangle &= \langle P_x\nabla f(x), \frac{\mathrm{dist}(x,x^*)}{\|P_x(x-x^*)\|}P_x(x-x^*)\rangle \\ &= \frac{\mathrm{dist}(x,x^*)}{\|P_x(x-x^*)\|}\langle P_x\nabla f(x), x-x^*\rangle.\end{aligned}$$

because $P_x^2 = P_x$. For the exact expressions for the gradient and the logarithm recall Section 1.3.1.1.

Direct calculation now gives

$$\begin{aligned}\langle P_x\nabla f(x), x-x^*\rangle &= -2x^TAx + 2\langle Ax, x^*\rangle - 2f(x)\|x\|^2 + 2f(x)\langle x, x^*\rangle \\ &= 2f(x) + 2\lambda_1\alpha_1 - 2f(x) + 2f(x)\alpha_1 \\ &= 2\alpha_1(f(x) + \lambda_1) = 2\alpha_1(f(x) - f^*) \geq 0.\end{aligned}$$

It is easy to verify that $\text{dist}(x, x^*) \geq \|P_x(x - x^*)\|$. We thus obtain

$$\langle \text{grad} f(x), -\text{Log}_x(x^*) \rangle \geq 2\alpha_1 (f(x) - f^*),$$

which gives the desired result since $\alpha_1 \geq a$. ■

We continue by providing a quadratic growth condition for our cost function, which can also be found in [116, Lemma 2] in a slightly different form. Here dist is the intrinsic distance in the sphere, that we also define in Section 1.3.1.1.

Proposition 3.2 *The function $f(x) = -x^T Ax$ satisfies*

$$f(x) - f^* \geq \frac{\mu}{2} \text{dist}^2(x, x^*), \; \mu := \frac{\delta}{2},$$

for any $x \in B_a$ with $a > 0$.

Proof The proof follows the one in [116, Lemma 2]. Using the expansions in (3.3.1.2), we get

$$x^T Ax = \sum_{i=1}^{n} \lambda_i \alpha_i^2 = \lambda_1 \alpha_1^2 + \sum_{i=2}^{n} \lambda_i \alpha_i^2 \leq \lambda_1 \alpha_1^2 + \lambda_2 (1 - \alpha_1^2)$$

since $\|x\|^2 = 1 = \sum_{i=1}^{n} \alpha_i^2$. From (1.3.1.2), we have that $\alpha_1 = \cos(\text{dist}(x, x^*))$ and so

$$x^T Ax \leq \lambda_1 \cos^2(\text{dist}(x, x^*)) + \lambda_2 \sin^2(\text{dist}(x, x^*)).$$

Direct calculation now shows

$$\begin{aligned} f(x) - f^* &= -x^T Ax + \lambda_1 \geq \lambda_1 - \lambda_1 \cos^2(\text{dist}(x, x^*)) - \lambda_2 \sin^2(\text{dist}(x, x^*)) \\ &= \lambda_1 \sin^2 \text{dist}(x, x^*)) - \lambda_2 \sin^2(\text{dist}(x, x^*)) = \delta \sin^2(\text{dist}(x, x^*)). \end{aligned}$$

Since $x \in B_a$ with $a > 0$, we have that x and x^* are in the same hemisphere and thus $d = \text{dist}(x, x^*) \leq \pi/2$. The desired result follows using $\sin(\phi) \geq \phi/2$ for $0 \leq \phi \leq \pi/2$. ■

Next, we prove that quadratic growth and weak-quasi convexity imply a WQSC property, similarly with Section 2.

Proposition 3.3 *f satisfies*

$$f(x) - f^* \leq \frac{1}{a} \langle \text{grad} f(x), -\text{Log}_x(x^*) \rangle - \frac{\mu}{2} \text{dist}^2(x, x^*),$$

for any $x \in B_a$, with $a > 0$.

Proof From quadratic growth and weak-quasi convexity, we have

$$\frac{\mu}{2}\text{dist}^2(x, x^*) \leq f(x) - f^* \leq \frac{1}{2a}\langle \text{grad} f(x), -\text{Log}_x(x^*)\rangle.$$

Now, again by weak-quasi convexity

$$\begin{aligned} f(x) - f^* &\leq \frac{1}{2a}\langle \text{grad} f(x), -\text{Log}_x(x^*)\rangle + \frac{\mu}{2}\text{dist}^2(x, x^*) - \frac{\mu}{2}\text{dist}^2(x, x^*) \\ &\leq \frac{1}{a}\langle \text{grad} f(x), -\text{Log}_x(x^*)\rangle - \frac{\mu}{2}\text{dist}^2(x, x^*) \end{aligned}$$

by substituting the previous inequality. ■

Interestingly, using Proposition 3.3, we can recover the PL property proved in [117, Theorem 4].

Proposition 3.4 *f satisfies*

$$\|\text{grad} f(x)\|^2 \geq \delta a^2 (f(x) - f^*)$$

for any $x \in B_a$ with $a > 0$.

Proof By Proposition 3.3, we have

$$f(x) - f^* \leq \frac{1}{a}\langle \text{grad} f(x), -\text{Log}_x(x^*)\rangle - \frac{\delta}{4}\text{dist}^2(x, x^*)$$

since, in our case, $\eta = \delta/2$. Using $\langle x, y\rangle \leq \frac{1}{2}(\|x\|^2 + \|y\|^2)$ for all $x, y \in \mathbb{R}^d$, we can write for any positive ρ that

$$\langle \text{grad} f(x), -\text{Log}_x(x^*)\rangle \leq \frac{\rho}{2}\|\text{grad} f(x)\|^2 + \frac{1}{2\rho}\|\text{Log}_x(x^*)\|^2.$$

Combining with $\rho = \frac{2}{a\delta}$ and using (1.3.1.2), we get

$$f(x) - f^* \leq \frac{1}{a}\frac{1}{a\delta}\|\text{grad} f(x)\|^2 + \frac{1}{a}\frac{a\delta}{4}\text{dist}^2(x, x^*) - \frac{\delta}{4}\text{dist}^2(x, x^*) = \frac{1}{a^2\delta}\|\text{grad} f(x)\|^2.$$

■

Proposition 3.5 *The function $f(x) = -x^T A x$ is geodesically $2(\lambda_1 - \lambda_n)$-smooth in the sphere.*

Proof The proof can be found also in [50, Lemma 1]. For $x \in \mathbb{S}^{n-1}$ and $v \in T_x\mathbb{S}^{n-1}$ with $\|v\| = 1$, we have that the Riemannian Hessian of f satisfies

$$\begin{aligned} \langle v, \nabla^2 f(x) v\rangle &= \langle v, -(I - xx^T)2Av + x^T 2Axv\rangle \\ &= -2v^T Av + 2x^T Ax \leq 2(\lambda_1 - \lambda_n) \end{aligned}$$

because $v^T A v \geq \lambda_n$ and $x^T A x \leq \lambda_1$, by the definition of eigenvalues and $\|x\|, |v\| = 1$. Similarly

$$-\langle v, \nabla^2 f(x) v \rangle = 2v^T A v - 2x^T A x \leq 2(\lambda_n - \lambda_1).$$

This finishes the proof, as the eigenvalues of $\nabla^2 f(x)$ are upper bounded by $2(\lambda_1 - \lambda_n)$ in absolute value. ■

Thus, the smoothness constant of f, as defined in Definition 1.19, equals $L = 2(\lambda_1 - \lambda_n)$. Similarly, let L_i denote the smoothness constant of f_i, which equals twice the difference of the largest eigenvalue to the smallest eigenvalue of A_i. In order to estimate L in a distributed fashion using only the local data matrices A_i, we shall use the over-approximation $L \leq 2p \max_{i=1,\dots,p} \lambda_{\max}(A_i)$.

3.3.2 Convergence

We now consider Riemannian gradient descent with learning rate $\eta > 0$ starting from a point $x_0 \in B_a$:

$$x_{t+1} = \mathrm{Exp}_{x_t}(-\eta \, \mathrm{grad} f(x_t)),$$

where Exp is the exponential map of the sphere, defined in Section 1.3.1.1.

Using Proposition 3.3 and a proper choice of η, we can establish a convergence rate for the instrinsic distance of the iterates to the minimizer.

Proposition 3.6 *An iterate of Riemannian gradient descent applied to $f(x) = -x^T A x$ starting from a point $x_t \in B_a$ and with step size $\eta \leq a/L$ where $L \geq 2(\lambda_1 - \lambda_n)$, produces a point x_{t+1} that satisfies*

$$\mathrm{dist}^2(x_{t+1}, x^*) \leq (1 - a\mu\eta) \, \mathrm{dist}^2(x_t, x^*) \qquad \text{for } \mu = \delta/2.$$

Note that this result implies directly that if our initialization x_0 lies in the ball B_a, the distance of x_1 to the center x^* decreases and thus all subsequent iterates continue being in the initialization ball. This is essential since it guarantees that the convexity-like properties for f continue to hold during the whole optimization process.

Proof By definition of x_{t+1}, we have $\mathrm{Log}_{x_t}(x_{t+1}) = -\eta \, \mathrm{grad} f(x_t)$. Applying Proposition 1.15, we can thus write

$$\begin{aligned} \mathrm{dist}^2(x_{t+1}, x^*) &\leq \| - \eta \, \mathrm{grad} f(x_t) - \mathrm{Log}_{x_t}(x^*)\|^2 \\ &= \eta^2 \|\mathrm{grad} f(x_t)\|^2 + \|\mathrm{Log}_{x_t}(x^*)\|^2 + 2\eta \langle \mathrm{grad} f(x_t), \mathrm{Log}_{x_t}(x^*) \rangle. \end{aligned}$$

By Propositions 3.3 and 1.21, we have

$$\begin{aligned} \frac{1}{a} \langle \mathrm{grad} f(x_t), \mathrm{Log}_{x_t}(x^*) \rangle &\leq f^* - f(x_t) - \frac{\mu}{2} \mathrm{dist}^2(x_t, x^*) \\ &\leq -\frac{1}{2\gamma} \|\mathrm{grad} f(x_t)\|^2 - \frac{\mu}{2} \mathrm{dist}^2(x_t, x^*). \end{aligned}$$

Multiplying with $2\eta a$ and using $\eta \leq a/\gamma$, we get

$$\begin{aligned} 2\eta\langle \mathrm{grad} f(x_t), \mathrm{Log}_{x_t}(x^*)\rangle &\leq -\frac{\eta a}{\gamma}\|\mathrm{grad} f(x_t)\|^2 - \mu\eta a \,\mathrm{dist}^2(x_t, x^*) \\ &\leq -\eta^2\|\mathrm{grad} f(x_t)\|^2 - \mu\eta a \,\mathrm{dist}^2(x_t, x^*). \end{aligned}$$

Substituting to the first inequality, we get the desired result. ■

3.4 Distributed gradient descent with limited communication

We now present our version of distributed gradient descent for leading eigenvector computation and measure its bit complexity until reaching accuracy ϵ in terms of the intrinsic distance of an iterate x_T from the minimizer x^* (x^* is the leading eigenvector closest to the initialization point).

Lattice quantization. For estimating the Riemannian gradient in a distributed manner with limited communication, we use a quantization procedure developed in [31]. The original quantization scheme involves randomness, but we use a *deterministic* version of it, by picking up the closest point to the vector that we want to encode. This is similar to the quantization scheme used by [63] and has the following properties.

Proposition 3.7 [63, 31] *Denoting by b the number of bits that each machine uses to communicate, there exists a quantization function*

$$Q\colon \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}^n,$$

which, for each $w, y > 0$, consists of an encoding function $\mathrm{enc}_{w,y} : \mathbb{R}^n \to \{0,1\}^b$ and a decoding one $\mathrm{dec}_{w,y} : \{0,1\}^b \times \mathbb{R}^n \to \mathbb{R}^n$, such that, for all $x, x' \in \mathbb{R}^n$,

- $\mathrm{dec}_{w,y}(\mathrm{enc}_{w,y}(x), x') = Q(x, x', y, w)$, *if* $\|x - x'\| \leq y$.
- $\|Q(x, x', y, w) - x\| \leq w$, *if* $\|x - x'\| \leq y$.
- *If $y/w > 1$, the cost of the quantization procedure in number of bits satisfies $b = \mathcal{O}\left(n\log(\frac{y}{w})\right)$.*

In the following, the quantization takes place in the tangent space of each iterate $T_{x_t}\mathbb{S}^{n-1}$, which is linearly isomorphic to $\mathbb{R}^{n-1}$. We denote by Q_x the specification of the function Q at $T_x\mathbb{S}^{n-1}$. The vector inputs of the function Q_x are represented in the local coordinate system of the tangent space that the quantization takes place at each step. For decoding at $t > 0$, we use information obtained in the previous step, that we need to translate to the same tangent space. We do that using parallel transport (see Section 1.3.1.1 for the formula).

Algorithm We present now our main algorithm, which is inspired by quantized gradient descent firstly designed by [74], and its similar version in [63]. The communication model is centralized in the sense that all nodes communicate their messages to a master node. The master node merges all the information, updates its eigenvector approximation and then sends it back to the rest of the nodes. For the rest, we use

$$L := 2p \max_{i=1,\dots,p} \lambda_{\max}(A_i).$$

---

1. Choose an arbitrary machine to be the master node, let it be i_0.
2. Choose $x_0 \in \mathbb{S}^{n-1}$ (we analyze later specific ways to do that).
3. Consider the following parameters

$$\sigma := 1 - \cos(D)\mu\eta,\ K := \frac{2}{\sqrt{\sigma}},\ \theta := \frac{\sqrt{\sigma}(1-\sqrt{\sigma})}{4},$$
$$\sqrt{\xi} := \theta K + \sqrt{\sigma},\ R_t = LK\left(\sqrt{\xi}\right)^t D$$

   where D is an over-approximation for $\operatorname{dist}(x_0, x^*)$.
   Assume that $\cos(D)\mu\eta \leq \frac{1}{2}$, otherwise run the algorithm with $\sigma = \frac{1}{2}$.
   In $T_{x_0}\mathbb{S}^{n-1}$:
4. Compute the local Riemannian gradient $\operatorname{grad} f_i(x_0)$ at x_0 in each node.
5. Encode $\operatorname{grad} f_i(x_0)$ in each node and decode in the master node using its local information:

$$q_{i,0} = Q_{x_0}\left(\operatorname{grad} f_i(x_0), \operatorname{grad} f_{i_0}(x_0), 4\lambda_1, \frac{\theta R_0}{2p}\right).$$

6. Sum the decoded vectors in the master node:

$$R_0 = \sum_{i=1}^{p} q_{i,0}.$$

7. Encode the sum in the master and decode in each machine i using its local information:

$$q_0 = Q_{x_0}\left(R_0, \operatorname{grad} f_i(x_0), \frac{\theta R_0}{2} + 4\lambda_1, \frac{\theta R_0}{2}\right).$$

   For $t \geq 0$:

8. Take a gradient step using the exponential map:

$$x_{t+1} = \mathrm{Exp}_{x_t}(-\eta q_t)$$

with step size η (the choice of step size is discussed later).
In $T_{x_{t+1}}\mathbb{S}^{n-1}$:

9. Compute the local Riemannian gradient $\mathrm{grad} f_i(x_{t+1})$ at x_{t+1} in each node.

10. Encode $\mathrm{grad} f_i(x_{t+1})$ in each node and decode in the master node using its (parallelly transported) local information from the previous step:

$$q_{i,t+1} = Q_{x_{t+1}}\left(\mathrm{grad} f_i(x_{t+1}), \Gamma_{x_t}^{x_{t+1}} q_{i,t}, \frac{R_{t+1}}{p}, \frac{\theta R_{t+1}}{2p}\right).$$

11. Sum the decoded vectors in the master node:

$$R_{t+1} = \sum_{i=1}^{p} q_{i,t+1}.$$

12. Encode the sum in the master and decode in each machine using its local information in the previous step after parallel transport:

$$q_{t+1} = Q_{x_{t+1}}\left(R_{t+1}, \Gamma_{x_t}^{x_{t+1}} q_t, \left(1+\frac{\theta}{2}\right) R_{t+1}, \frac{\theta R_{t+1}}{2}\right).$$

Convergence We first control the convergence of iterates simultaneously with the convergence of quantized gradients.

Note that

$$\sqrt{\xi} = \frac{1-\sqrt{\sigma}}{2} + \sqrt{\sigma} = \frac{1+\sqrt{\sigma}}{2} \leq \frac{\sqrt{2}\sqrt{1+\sigma}}{2} = \sqrt{\frac{1+\sigma}{2}}.$$

Lemma 3.8 *If $\eta \leq \cos(D)/L$, the previous quantized gradient descent algorithm produces iterates x_t and quantized gradients q_t that satisfy*

$$(i)\mathrm{dist}^2(x_t, x^*) \leq \xi^t D^2,\ (ii)\|q_{i,t} - \mathrm{grad} f_i(x_t)\| \leq \frac{\theta R_t}{2p},\ (iii)\|q_t - \mathrm{grad} f(x_t)\| \leq \theta R_t.$$

The proof is a Riemannian adaptation of the similar one in [74] and [63]. We recall that since the sphere is positively curved, it provides a landscape easier for optimization. It is quite direct to derive a general Riemannian method for manifolds of bounded curvature using more advanced geometric

bounds, however this exceeds the scope of this section, which focuses on leading eigenvector computation.

Proof We do the proof by induction. We start from the case that $t = 0$. (i) is direct by the definition of D.

For (ii), we have

$$\|\text{grad}f_i(x_0) - \text{grad}f_{i_0}(x_0)\| \leq \|\text{grad}f_i(x_0)\| + \|\text{grad}f_{i_0}(x_0)\| \leq 4\lambda_1.$$

This is because $\|\text{grad}f_i(x_0)\| = \|2P_{x_0}A_ix_0\| \leq 2\|A_ix_0\| \leq 2\lambda_{\max}(A_i) \leq 2\lambda_1$, since $A = \sum_{i=1}^{p} A_i$ and all A_i's are positive semi-definite. Similarly for $\|\text{grad}f_{i_0}(x_0)\|$.

By the definition of quantization (step 5), we get

$$\|\text{grad}f_i(x_0) - q_{i,0}\| \leq \frac{\theta R_0}{2p}.$$

Similarly for (iii), we have

$$\|\text{grad}f(x_0) - R_0\| \leq \sum_{i=1}^{p} \|\text{grad}f_i(x_0) - q_{i,0}\| \leq \frac{\theta R_0}{2}.$$

Then,

$$\|R_0 - \text{grad}f_i(x_0)\| \leq \|R_0 - \text{grad}f(x_0)\| + \|\text{grad}f(x_0) - \text{grad}f_i(x_0)\| \leq \frac{\theta R_0}{2} + 4\lambda_1.$$

By the definition of the quantization (step 7), we have

$$\|q_0 - R_0\| \leq \frac{\theta R_0}{2}.$$

Thus,

$$\|q_0 - \text{grad}f(x_0)\| \leq \|\text{grad}f(x_0) - R_0\| + \|q_0 - R_0\| \leq \frac{\theta R_0}{2} + \frac{\theta R_0}{2} = \theta R_0.$$

We assume now that the inequalities hold for t and we wish to prove that they continue to hold for $t + 1$.

We start with (i) and denote by $\tilde{x}_{t+1}$ the iteration of exact gradient descent starting from x_t. Since $\text{dist}(x_t, x^*) \leq D$, we have that $x_t \in B_a$ with $a = \cos(D)$.

We have

$$\begin{aligned}\text{dist}(x_{t+1}, x^*) &\leq \text{dist}(x_{t+1}, \tilde{x}_{t+1}) + \text{dist}(\tilde{x}_{t+1}, x^*) \\ &\leq \|\eta\text{grad}f(x_t) - \eta q_t\| + \sqrt{\sigma}\text{dist}(x_t, x^*).\end{aligned}$$

We have the last inequality, because

$$\text{dist}(\tilde{x}_{t+1}, x^*) \leq \sqrt{\sigma}\text{dist}(x_t, x^*)$$

by Proposition 3.6 and

$$\operatorname{dist}(x_{t+1}, \tilde{x}_{t+1}) \leq \| \operatorname{Log}_{x_t}(x_{t+1}) - \operatorname{Log}_{x_t}(\tilde{x}_{t+1})\| = \|\eta \operatorname{grad} f(x_t) - \eta q_t\|$$

by Proposition 1.15.

Thus

$$\begin{aligned}\operatorname{dist}(x_{t+1}, x^*) &\leq \frac{a}{L}\theta R_t + \sqrt{\sigma}\left(\sqrt{\xi}\right)^t D \leq \theta K \left(\sqrt{\xi}\right)^t D + \sqrt{\sigma}\left(\sqrt{\xi}\right)^t D \\ &\leq (\theta K + \sqrt{\sigma})\left(\sqrt{\xi}\right)^t D \leq \left(\sqrt{\xi}\right)^{t+1} D\end{aligned}$$

which concludes the induction for the first inequality.

For (ii), we have

$$\begin{aligned}\|\operatorname{grad} f_i(x_{t+1}) - \Gamma_{x_t}^{x_{t+1}} q_{i,t}\| &\leq \|\operatorname{grad} f_i(x_{t+1}) - \Gamma_{x_t}^{x_{t+1}} \operatorname{grad} f_i(x_t)\| \\ &\quad + \|\Gamma_{x_t}^{x_{t+1}} \operatorname{grad} f_i(x_t) - \Gamma_{x_t}^{x_{t+1}} q_{i,t}\| \\ &\leq L_i \operatorname{dist}(x_{t+1}, x_t) + \|\operatorname{grad} f_i(x_t) - q_{i,t}\| \\ &\leq 2\frac{L}{p}\left(\sqrt{\xi}\right)^t D + \theta \frac{R_t}{p} \\ &= 2\frac{L}{p}\left(\sqrt{\xi}\right)^t D + \theta L K \left(\sqrt{\xi}\right)^t D/p \\ &= (2/K + \theta) K L \left(\sqrt{\xi}\right)^t D/p \\ &\leq (\sqrt{\sigma} + \theta K) K L \left(\sqrt{\xi}\right)^t D/p \\ &= \frac{R_{t+1}}{p}\end{aligned}$$

and by the definition of the quantization scheme (step 10), we have

$$\|\operatorname{grad} f_i(x_{t+1}) - q_{i,t+1}\| \leq \frac{\theta R_{t+1}}{2p}.$$

For (iii), we have

$$\|R_{t+1} - \operatorname{grad} f(x_{t+1})\| \leq \sum_{i=1}^{n} \|q_{i,t+1} - \operatorname{grad} f_i(x_{t+1})\| \leq \frac{\theta R_{t+1}}{2}$$

and

$$\begin{aligned}\|R_{t+1} - \Gamma_{x_t}^{x_{t+1}} q_t\| &\leq \|R_{t+1} - \operatorname{grad} f(x_{t+1})\| + \|\operatorname{grad} f(x_{t+1}) - \Gamma_{x_t}^{x_{t+1}} \operatorname{grad} f(x_t)\| \\ &\quad + \|\Gamma_{x_t}^{x_{t+1}} \operatorname{grad} f(x_t) - \Gamma_{x_t}^{x_{t+1}} q_t\| \\ &\leq \frac{\theta R_{t+1}}{2} + L \operatorname{dist}(x_{t+1}, x_t) + \theta R_t \\ &\leq \frac{\theta R_{t+1}}{2} + R_{t+1} = \left(1 + \frac{\theta}{2}\right) R_{t+1}\end{aligned}$$

by using again the argument for deriving the second inequality. The last inequality implies that

$$\|R_{t+1} - q_{t+1}\| \leq \frac{\theta R_{t+1}}{2}$$

by the definition of quantization (step 12). We can now write

$$\begin{aligned}\|q_{t+1} - \mathrm{grad} f(X_{t+1})\| &\leq \|q_{t+1} - R_{t+1}\| + \|R_{t+1} - \mathrm{grad} f(X_{t+1})\| \\ &\leq \frac{\theta R_{t+1}}{2} + \frac{\theta R_{t+1}}{2} = \theta R_{t+1}.\end{aligned}$$

This completes the induction. ■

We now move to our main complexity result.

Theorem 3.9 *Let $\eta \leq \cos(D)/L$. Then, the previous quantized gradient descent algorithm needs at most*

$$b = \mathcal{O}\left(pn \frac{1}{\cos(D)\delta\eta} \log\left(\frac{p}{\cos(D)\delta\eta}\right) \log\left(\frac{D}{\epsilon}\right)\right) = \tilde{\mathcal{O}}\left(\frac{pn}{\cos(D)\delta\eta}\right)$$

bits in total to estimate the leading eigenvector with an accuracy ϵ measured in intrinsic distance.

The proof is based on the previous Lemma 3.8 in order to count the number of steps that the algorithm needs to estimate the minimizer with accuracy ϵ and Proposition 3.7 to count the quantization cost in each round.

Proof For computing the cost of quantization at each step, we use Proposition 3.7.

The communication cost of encoding each $\mathrm{grad} f_i$ at $t = 0$

$$\mathcal{O}\left(n \log \frac{4\lambda_1}{\frac{\theta R_0}{2p}}\right) = \mathcal{O}\left(n \log \frac{8p\lambda_1}{\theta LKD}\right) \leq \mathcal{O}\left(n \log \frac{2p}{\theta D}\right).$$

This is because $2\lambda_1 \leq L$.

Now we use that $\sigma \geq \frac{1}{2}$ and have

$$\frac{1}{\theta} = \frac{4}{\sqrt{\sigma}(1 - \sqrt{\sigma})} \leq \frac{12}{1 - \sigma} = \frac{12}{\cos(D)\mu\eta}.$$

Thus, the previous cost becomes

$$\mathcal{O}\left(n \log \frac{4\lambda_1}{\frac{\theta R_0}{2p}}\right) = \mathcal{O}\left(n \log \frac{p}{D\cos(D)\mu\eta}\right).$$

As D is only an over-approximation for the initial distance, we can write this cost as

$$\mathcal{O}\left(n \log \frac{p}{\cos(D)\mu\eta}\right).$$

The communication cost of decoding each $q_{i,0}$ in the master node is

$$\mathcal{O}\left(n\,\mathrm{Log}\,\frac{4\lambda_1+\frac{\theta R_0}{2}}{\frac{\theta R_0}{2}}\right)\leq\mathcal{O}\left(n\log\frac{4\lambda_1}{\frac{\theta R_0}{2}}\right)\leq\mathcal{O}\left(n\log\frac{1}{\cos(D)\mu\eta}\right).$$

Thus, the total communication cost at $t=0$ is

$$\mathcal{O}\left(pn\log\frac{p}{\cos(D)\mu\eta}\right).$$

For $t>0$, the cost of encoding $\mathrm{grad} f_i$'s is

$$\mathcal{O}\left(pn\log\frac{R_{t+1}/p}{\theta R_{t+1}/2p}\right)=\mathcal{O}\left(pn\,\mathrm{Log}\,\frac{2}{\theta}\right)=\mathcal{O}\left(pn\log\frac{1}{\cos(D)\mu\eta}\right).$$

as before.

The cost of decoding in the master node is

$$\mathcal{O}\left(pn\log\frac{(1+\theta/2)R_{t+1}}{\theta R_{t+1}/2}\right)\leq\mathcal{O}\left(pn\,\mathrm{Log}\,\frac{1}{\theta}\right)=\mathcal{O}\left(pn\log\frac{1}{\cos(D)\mu\eta}\right).$$

Thus, the cost in each round of communication is in general bounded by

$$\mathcal{O}\left(pn\log\frac{p}{\cos(D)\mu\eta}\right).$$

Our algorithm reaches accuracy ϵ in function values if

$$\mathrm{dist}(x_t,x^*)\leq\epsilon.$$

We can now write

$$\mathrm{dist}^2(x_t,x^*)\leq\xi^tD^2\leq e^{-(1-\xi)t}D^2.$$

Thus, we need to run our algorithm for

$$\mathcal{O}\left(\frac{1}{1-\xi}\log\frac{D}{\epsilon}\right)\leq\mathcal{O}\left(\frac{1}{\cos(D)\mu\eta}\log\frac{D}{\epsilon}\right)$$

many iterates to reach accuracy ϵ.

The total communication cost for doing that is

$$\mathcal{O}\left(\frac{1}{\cos(D)\mu\eta}\log\left(\frac{D}{\epsilon}\right)pn\log\left(\frac{p}{\cos(D)\mu\eta}\right)\right)=$$
$$\mathcal{O}\left(pn\frac{1}{\cos(D)\mu\eta}\log\left(\frac{p}{\cos(D)\mu\eta}\right)\log\left(\frac{D}{\epsilon}\right)\right)$$

Substituting

$$\mu = \frac{\delta}{2}$$

by Proposition 3.2, we get

$$\mathcal{O}\left(pn\frac{1}{\cos(D)\delta\eta}\log\left(\frac{p}{\cos(D)\delta\eta}\right)\log\left(\frac{D}{\epsilon}\right)\right)$$

many bits in total. ■

3.5 Dependence on initialization

3.5.1 Uniformly random initialization

The cheapest choice to initialize quantized gradient descent is a point in the sphere chosen uniformly at random. According to Theorem 4 in [117], such a random point x_0 will lie, with probability at least $1 - \mathrm{pr}$, in a ball B_a (see (3.3.1.1)) where

$$a \geq c\frac{\mathrm{pr}}{\sqrt{n}} \iff \mathrm{dist}(x_0, x^*) \leq \arccos\left(c\frac{n}{\sqrt{\mathrm{pr}}}\right). \tag{3.5.1.1}$$

Here, c is a universal constant. We estimated numerically that c is around 1.25 (see Figure 3.1), thus we can use $c = 1$ for simplicity.

Let x_0 be chosen from a uniform distribution in the sphere $\mathbb{S}^{n-1}$. We are interested in $\alpha_1 = v_1^T x_0$ for some fixed $v_1 \in \mathbb{S}^{n-1}$. By spherical symmetry, α_1 is distributed in the same way as the first component of x_0. Let $A_n(h)$ be the surface of the hyperspherical cap of $\mathbb{S}^{n-1}$ with height $h \in [0, 1]$. Then it is obvious that

$$\mathbb{P}(|\alpha_1| \geq a) = A_n(1-a)/A_n(1) = I_{1-a^2}(\tfrac{n-1}{2}, \tfrac{1}{2}),$$

where we used the well-known formula for $A_n(h)$ in terms of the regularized incomplete Beta function $I_x(a, b)$; see, e.g., [70]. Solving the above expression[8] for a when it equals a given probability $1 - \mathrm{pr}$, we can calculate the interval $[-1, -a] \cup [a, 1]$ in which α_1 will lie for a random x_0 up to probability $1 - \mathrm{pr}$.

In the figure below, we have plotted these values of a divided by $\mathrm{pr}/\sqrt{n}$ for $\mathrm{pr} = 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}$. Numerically, there is strong evidence that $a \geq c\frac{\mathrm{pr}}{\sqrt{n}}$ with $c \approx 1.25$.

[8] This can be conveniently done using `https://docs.scipy.org/doc/scipy/reference/generated/scipy.special.betaincinv.html`

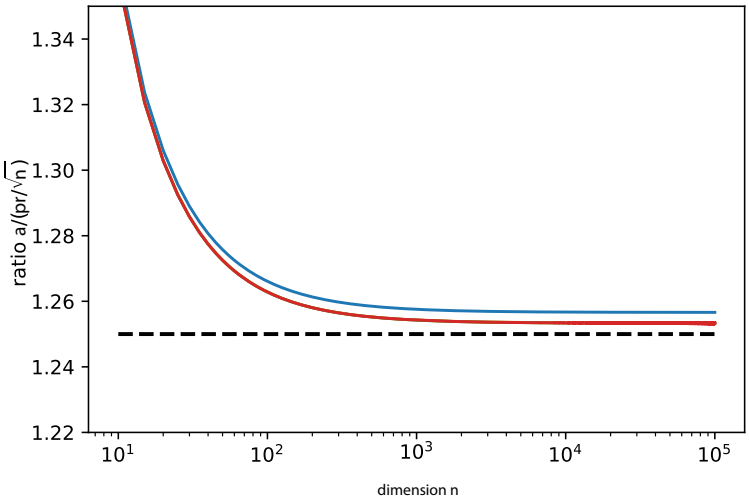


Figure 3.1: Numerical estimation of the constant c.

By choosing the step-size η as

$$\eta = \frac{c \cdot \mathrm{pr}}{\sqrt{n}L}$$

and the parameter D as

$$D = \arccos\left(c\frac{\mathrm{pr}}{\sqrt{n}}\right),$$

we are guaranteed that $\eta = \frac{\cos(D)}{L}$, and $\mathrm{dist}(x_0, x^*) \leq D$ with probability at least $1 - \mathrm{pr}$. Our analysis above therefore applies (up to probability $1 - \mathrm{pr}$) and the general communication complexity result becomes

$$\begin{aligned}\mathcal{O}\left(\frac{pn}{\cos(D)\delta\eta}\log\frac{p}{\cos(D)\delta\eta}\log\frac{D}{\epsilon}\right) &= \mathcal{O}\left(\frac{pn}{\eta L\delta\eta}\log\frac{p}{\eta L\delta\eta}\log\frac{D}{\epsilon}\right)\\ &= \mathcal{O}\left(\frac{pn}{\eta^2 L\delta}\log\frac{p}{\eta^2 L\delta}\log\frac{D}{\epsilon}\right).\end{aligned}$$

Substituting $\eta^2 = \frac{\mathrm{pr}^2 c^2}{nL^2}$, the number of bits satisfies (up to probability $1 - \mathrm{pr}$) the upper bound

$$b = \mathcal{O}\left(p\frac{n^2}{\mathrm{pr}^2}\frac{L}{\delta}\log\frac{pnL}{\mathrm{pr}\delta}\log\frac{D}{\epsilon}\right) = \tilde{\mathcal{O}}\left(p\frac{n^2}{\mathrm{pr}^2}\frac{L}{\delta}\right).$$

3.5.2 Warm start

A reasonable strategy to get a more accurate initialization is to perform an eigenvalue decomposition to one of the local covariance matrices, for instance A_{i_0} (in the master node i_0), and compute its leading eigenvector, let it be v_{i_0}. For simplicity we will assume here that each machine hosts the same number of data points $m_i = \frac{m}{p}$ (assuming of course that p divides m exactly). Then, we communicate v_{i_0} to all machines in order to use the normalized quantized approximation x_0 as initialization. We define:

$$\tilde{x}_0 = Q\left(v_{i_0}, v_i, \|v_{i_0} - v_i\|, \frac{\langle v_{i_0}, x^* \rangle}{2(\sqrt{2}+2)}\right)$$
$$x_0 = \frac{\tilde{x}_0}{\|\tilde{x}_0\|}$$

where Q is the lattice quantization scheme in $\mathbb{R}^n$ (i.e. we quantize the leading eigenvector of the master node as a vector in $\mathbb{R}^n$ and then project back to the sphere). The input and output variance in this quantization can be bounded by constants that we can practically estimate.

Proposition 3.10 *Assume that our data are i.i.d. and sampled from a distribution $\mathcal{D}$ bounded in ℓ_2 norm by a constant h. Given that the spectral gap δ, the number of machines p and the total number of data points m satisfy*

$$\delta \geq \Omega\left(\sqrt{m}\sqrt{p}\sqrt{\log\frac{n}{\mathrm{pr}}}\right), \tag{3.5.2.1}$$

we have that the previous quantization costs $\mathcal{O}(pn)$ many bits and $\langle x_0, x^* \rangle$ is lower bounded by some constant with probability at least $1 - \mathrm{pr}$.

Proof By Lemma 3 in [50] we have that

$$\left\| A_{i_0} - \frac{1}{m}\sum_{i=1}^{p} A_i \right\|^2 \leq \frac{32\log\left(\frac{n}{\mathrm{pr}}\right)h^2}{m_{i_0}}$$

which implies that

$$\left\| mA_{i_0} - \sum_{i=1}^{p} A_i \right\|^2 \leq 32\frac{m^2}{m_{i_0}}\log\left(\frac{n}{\mathrm{pr}}\right)h^2 = 32mp\log\left(\frac{n}{\mathrm{pr}}\right)h^2$$

with probability at least $1 - \mathrm{pr}$. Of course $\sum_{i=1}^{p} A_i = A$.

From this bound we can derive a bound for the distance between the eigenvectors of the two matrices. Indeed, using Lemmas 5 and 8 in [50], we can derive

$$1 - \langle v_{i_0}, x^* \rangle \leq \frac{\sqrt{128mp\log\left(\frac{n}{\mathrm{pr}}\right)}h}{\delta}$$

and

$$\langle v_{i_0}, x^* \rangle \geq 1 - \frac{\sqrt{128mp \log\left(\frac{n}{\mathrm{pr}}\right)} h}{\delta}$$

with probability at least $1 - \mathrm{pr}$ (note that the leading eigenvector of A_{i_0} is equal to the leading eigenvector mA_{i_0}). This is because $\langle v_{i_0}, x^* \rangle \leq 1$, which implies that $\langle v_{i_0}, x^* \rangle^2 \leq \langle v_{i_0}, x^* \rangle$.

We notice that the squared distance of v_{i_0} and x^* is

$$\|v_{i_0} - x^*\|^2 = \|v_{i_0}\|^2 + \|x^*\|^2 - 2\langle v_{i_0}, x^* \rangle = 2(1 - \langle v_{i_0}, x^* \rangle) \leq 2\frac{\sqrt{128mp \log\left(\frac{n}{\mathrm{pr}}\right)} h}{\delta}$$

which is upper bounded by a constant by Assumption (3.5.2.1). The same holds for $\|v_i - x^*\|$, thus, by the triangle inequality, we have an upper bound on $\|v_{i_0} - v_i\|$ to be at most double of the upper bound for $\|v_{i_0} - x^*\|$, thus it is still upper bounded by a constant. Since $\langle v_{i_0}, x^* \rangle$ is lower bounded by a constant, again by Assumption (3.5.2.1), we have that the ratio of the input to the output variance in the quantization of v_{i_0} is upper bounded by a constant. Thus, the total communication cost of this quantization is $\mathcal{O}(pn)$.

By the definition of the quantization scheme, we get

$$\|\tilde{x}_0 - v_{i_0}\| \leq \frac{\langle v_{i_0}, x^* \rangle}{2(\sqrt{2} + 2)} =: \tau.$$

For the projected vector x_0, we have

$$\|x_0 - v_{i_0}\| \leq \|\tilde{x}_0 - v_{i_0}\| + \|\tilde{x}_0 - x_0\| \leq 2\|\tilde{x}_0 - v_{i_0}\| \leq 2\tau$$

because x_0 is the closest point to $\tilde{x}_0$ belonging to the sphere and v_{i_0} belongs also to the sphere.

By the triangle inequality, we have

$$\|x_0 - x^*\| \leq \|v_{i_0} - x^*\| + \|x_0 - v_{i_0}\|$$

which is equivalent to

$$\sqrt{2(1 - \langle x_0, x^* \rangle)} \leq \sqrt{2(1 - \langle v_{i_0}, x^* \rangle)} + 2\tau.$$

Thus

$$\begin{aligned}\langle x_0, x^* \rangle &\geq \langle v_{i_0}, x^* \rangle - \sqrt{2(1 - \langle v^{i_0}, x^* \rangle)}\tau - 2\tau^2 \geq \langle v_{i_0}, x^* \rangle - \sqrt{2}\tau - 2\tau \\ &= \langle v_{i_0}, x^* \rangle - (\sqrt{2} + 2)\tau = \langle v_{i_0}, x^* \rangle - (\sqrt{2} + 2)\frac{\langle v_{i_0}, x^* \rangle}{2(\sqrt{2} + 2)} = \frac{\langle v_{i_0}, x^* \rangle}{2}\end{aligned}$$

with probability at least $1 - \mathrm{pr}$.

Since $\langle v_{i_0}, x^* \rangle$ is lower bounded by a constant, $\langle x_0, x^* \rangle$ is also lower bounded by a constant and we get the desired result.

■

Thus, if bound (3.5.2.1) is satisfied, then the communication complexity becomes

$$b = \mathcal{O}\left(pn\frac{L}{\delta}\log\frac{nL}{\delta}\log\frac{1}{\epsilon}\right) = \tilde{\mathcal{O}}\left(pn\frac{L}{\delta}\right)$$

many bits in total with probability at least $1 - \text{pr}$ (notice that this can be further simplified using bound (3.5.2.1)). This is because D in Theorem 3.9 is upper bounded by a constant and the communication cost of quantizing v_{i_0} does not affect the total communication cost. If we can estimate the specific relation in bound (3.5.2.1) (the constant hidden inside Ω), then we can compute estimations of the quantization parameters in the definition of x_0.

Condition (3.5.2.1) is quite typical in this literature; see [50] and references therein (beginning of page 2), as we also briefly discussed in the introduction. Notice that $\sqrt{m}$ appears in the numerator and not the denominator, only because we deal with the sum of local covariance matrices and not the average, thus our spectral gap is m times larger than the spectral gap of the normalized covariance matrix. Denoting by δ' the spectral gap of the normalized covariance matrix, bound (3.5.2.1) can be written equivalently as

$$\delta' \geq \Omega\left(\frac{\sqrt{p}}{\sqrt{m}}\sqrt{\log\frac{n}{\text{pr}}}\right) = \tilde{\Omega}\left(\frac{1}{\sqrt{m_{i_0}}}\right),$$

where m_{i_0} is the number of data points owned by the master node (and any other machine) and $\tilde{\Omega}$ hides logarithmic factors from the lower bound.

3.6 Numerical experiments

We evaluate our approach experimentally, comparing the proposed method of Riemannian gradient quantization against three other benchmark methods:

- Full-precision Riemannian gradient descent: Riemannian gradient descent, as described in Section 3.3.2, is performed with the vectors communicated at full (64-bit) precision.
- Euclidean gradient difference quantization: the "naïve" approach to quantizing Riemannian gradient descent. Euclidean gradients are quantized and averaged before being projected to Riemannian gradients and used to take a step. To improve performance, rather than quantizing Euclidean gradients directly, we quantize the difference between the current local

gradient and the previous local gradient, at each node. Since these differences are generally smaller than the gradients themselves, we expect this quantization to introduce lower error.

- Quantized power iteration: we also use as a benchmark a quantized version of power iteration, a common method for leading-eigenvector computation given by the update rule $x_{t+1} \leftarrow \frac{Ax_t}{\|Ax_t\|}$. Ax_t can be computed in distributed fashion by communicating and summing the vectors $A_i x_t, i \leq n$. It is these vectors that we quantize.

All three of the quantized methods use the same vector quantization routine, for fair comparison.

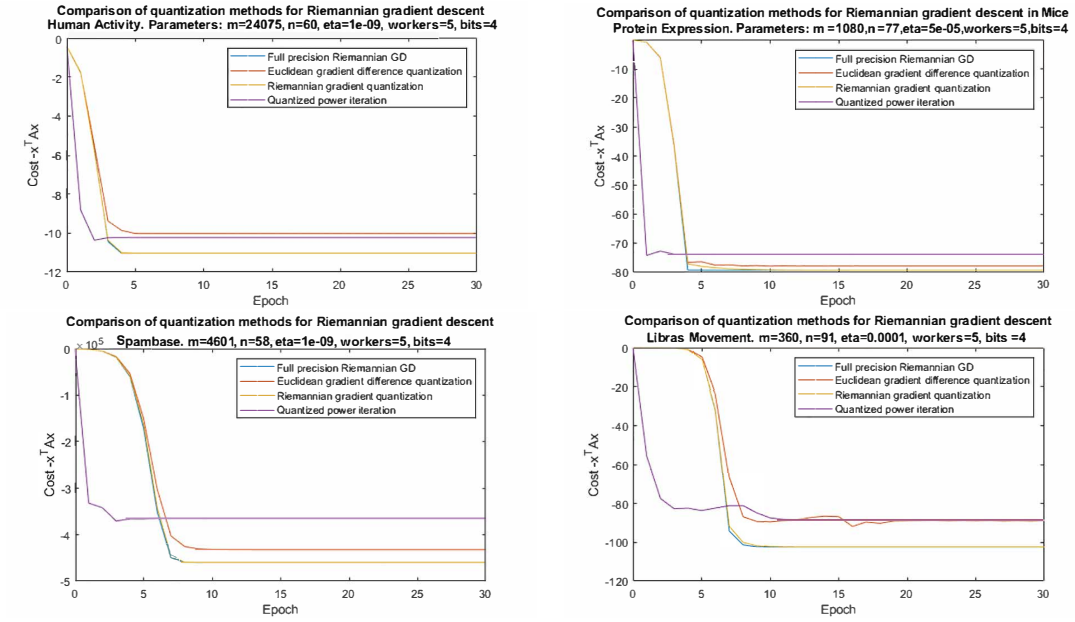


Figure 3.2: Convergence results on real datasets

We show convergence results (Figure 3.2) for the methods on four real datasets: Human Activity from the MATLAB Statistics and Machine Learning Toolbox, and Mice Protein Expression, Spambase, and Libras Movement from the UCI Machine Learning Repository [34]. All results are averages over 10 runs of the cost function $-x^T Ax$ (for each method, the iterates x are normalized to lie in the 1-ball, so a lower value of $-x^T Ax$ corresponds to being closer to the principal eigenvector).

All four datasets display similar behavior: our approach of Riemannian gradient quantization outperforms naïve Euclidean gradient quantization, and essentially matches the performance of the full-precision method while communicating only 4 bits per coordinate, for a 16× compression. Power iteration

converges slightly faster (as would be expected from the theoretical convergence guarantees), but is much more adversely affected by quantization, and reaches a significantly suboptimal result. Our code is publicly available [9].

[9] `https://github.com/IST-DASLab/QRGD`

4 Preconditioned inverse eigenvalue solvers

This section is concerned with novel results in the theory of preconditioned eigenvalue solvers. It follows the exposition of our work [9].

4.1 Introduction

We start by giving a general overview of preconditioned eigenvalue solvers. Given a large-scale, symmetric positive definite (SPD) matrix $A \in \mathbb{R}^{n\times n}$ with eigenvalues $0 < \lambda_1 < \lambda_2 \leq \cdots \leq \lambda_n$, this section considers the task of approximating the smallest eigenvalue λ_1 and an associated eigenvector u^*. Notice the difference in notations here: we target the *smallest* eigenvalue, which we denote λ_1 and use the symbol u^* (and not x^*) for its associated eigenvector (x^* is reserved for the optimum of our future objective optimization problem). Targeting the smallest or the largest eigenvalue is mathematically the same. Inverse iteration [39, Sec. 8.2.2] addresses this task by applying the power method to the inverse: $u_{t+1} = A^{-1}u_t$, combined with some normalization to avoid numerical issues. This iteration inherits the excellent global convergence guarantee of the power method [39, Thm. 8.2.1]: For *almost every* choice of starting vector u_0, the angle between u^* and u_t converges linearly to zero with rate λ_1/λ_2. Moreover, the Rayleigh quotient $\lambda(u_t) := u_t^T Au_t/u_t^T u_t$ converges linearly to λ_1 with rate λ_1^2/λ_2^2. A discussion about the convergence of power method can be found also in Section 1.2. Notice here again a difference in notation compared to Section 3: the Rayleigh quotient in the Euclidean space is denoted by $\lambda(u)$, while the name f is reserved for our future objective cost function.

A major limitation of the inverse iteration is that it requires to solve a linear system with A in every iteration. Using, for example, a sparse Cholesky factorization of A for this purpose may become expensive unless A has a favorable sparsity pattern. In many situations, it is much cheaper to apply B^{-1} instead of A^{-1} for a preconditioner B constructed, for example, from multigrid methods [24], domain decomposition [109] or spectral sparsification [66]. In principle, the availability of a preconditioner allows for the use of an iterative solver, such as the preconditioned conjugate gradient method [39, Sec. 11.5.2], for solving the linear systems with A within inverse iteration. However, combining iterative methods in such an inner-outer iteration typically incurs redundancies. Instead, it is preferable to incorporate the preconditioner more directly, in a *preconditioned eigenvalue solver*.

The fruitfly of preconditioned eigenvalue solvers is the Preconditioned INVerse ITeration (PINVIT)

$$u_{t+1} = u_t - B^{-1}r_t \quad \text{with} \quad r_t = Au_t - \lambda(u_t)u_t. \tag{4.1.0.1}$$

While PINVIT can be viewed as a preconditioned gradient descent method

for minimizing the Rayleigh quotient, Neymeyr's seminal (non-asymptotic) convergence analysis [82, 83] is based on interpreting (4.1.0.1) as a perturbed inverse iteration. *Assuming* that $\lambda(u_t) \in [\lambda_1, \lambda_2)$, a convergence result by Knyazev and Neymeyr [61, Thm. 1] states that

$$\frac{\lambda(u_{t+1}) - \lambda_1}{\lambda_2 - \lambda(u_{t+1})} \leq \alpha^2 \frac{\lambda(u_t) - \lambda_1}{\lambda_2 - \lambda(u_t)}, \tag{4.1.0.2}$$

with the convergence rate determined by $\alpha = 1 - (1 - \alpha_B)(1 - \lambda_1/\lambda_2)$, where α_B is such that

$$\|I - B^{-1}A\|_A \leq \alpha_B < 1. \tag{4.1.0.3}$$

Here and in the following, $\|\cdot\|_C$ denotes the vector and operator norms induced by an SPD matrix C. If $\alpha_B \ll 1$, this result shows that PINVIT nearly attains the convergence rate of inverse iteration. In principle, the condition (4.1.0.3) can always be satisfied for any SPD matrix B by suitably rescaling B to B/η, which is equivalent to adding a step size $\eta > 0$ to PINVIT: $u_{t+1} = u_t - \eta B^{-1} r_t$. With PINVIT being one of the simplest preconditioned eigenvalue solvers, its analysis also provides important insights into the performance of more advanced methods like LOBPCG [59] and PRIMME [107]. Recently, provable accelerations of PINVIT, in the sense of Nesterov's accelerated gradient descent [79], have been introduced in [102, 103], based on certain convexity structures of the Rayleigh quotient. The analysis of these methods requires conditions on the initial vector that are even stricter than the one required for PINVIT.

If $\lambda(u_0) \in [\lambda_1, \lambda_2)$ then (4.1.0.1) implies that $\lambda(u_t) \in [\lambda_1, \lambda_2)$ is satisfied for all subsequent iterates u_t of PINVIT and u_t converges to u^* (in terms of angles). However, this assumption on the initial vector u_0 is quite restrictive. In fact, for a Gaussian random initial u_0, the probability of achieving $\lambda(u_0) < \lambda_2$ quickly vanishes for larger n and does not benefit from the quality of the preconditioner B. This is in stark contrast to both, inverse iteration ($B = A$) and gradient descent [8] ($B = I$), which converge to u^* almost surely for a Gaussian random initial vector.

In this section, we present a new non-asymptotic convergence result for a slight variation of PINVIT. For this purpose, we first reformulate the task of computing the smallest eigenvalue and eigenvector as an equivalent Riemannian optimization problem on the unit sphere $\mathbb{S}^{n-1}$ in $\mathbb{R}^n$, with the preconditioner B incorporated. A similar but different reformulation was used in [102]. We show that standard Riemannian gradient descent [104, Algorithm 3.1] applied to this problem coincides with a variant of PINVIT (4.1.0.1) that uses a different step size and normalization. Moreover, we show that this problem has a WQSC structure, inspired by the results presented in Section 2, and in Section 3 for the case of computing one eigenvector. This yields as to our main result (Theorem 4.8): Riemannian gradient descent and, hence, our variant of PINVIT

converges if the initial vector u_0 satisfies

$$\frac{u_0^T B u^*}{\|u_0\|_B \|u^*\|_B} > \cos\varphi, \tag{4.1.0.4}$$

where φ is an angle measuring the distortion of the Euclidean geometry induced by the preconditioner at u^*; see (4.3.2.1) for the precise definition. The convergence is linear and we prove an asymptotic convergence rate that matches (4.1.0.2) up to a small factor.

For $B = I$ and $B = A$, it holds that $\cos\varphi = 0$ and, thus, the condition (4.1.0.4) recovers the excellent global convergence properties of gradient descent and inverse iteration mentioned above. The practical use of PINVIT is between these two extreme scenarios and in such cases our numerical results indicate that the condition (4.1.0.4) is less stringent than $\lambda(u_0) < \lambda_2$. For the specific choices of mixed-precision and domain decomposition preconditioners, we provide theoretical results underlining that good preconditioners lead to $\cos\varphi \approx 0$.

4.2 PINVIT as gradient descent

The results of this section are based on a novel formulation of PINVIT as (Riemannian) gradient descent on $\mathbb{S}^{n-1}$. For this purpose, we define the following optimization problem for SPD matrices $A, B \in \mathbb{R}^{n\times n}$:

$$\min_{x\in\mathbb{S}^{n-1}} f(x), \qquad f(x) := -\frac{x^T B^{-1} x}{x^T B^{-1/2} A B^{-1/2} x}. \tag{4.2.0.1}$$

Using the substitution $u = B^{-1/2}x$, we have that

$$f(x) = -\frac{u^T u}{u^T A u}.$$

The minimum of f is hence $-1/\lambda_1$ and is attained at $x^* = \frac{B^{1/2}u^*}{\|B^{1/2}u^*\|}$ for an eigenvector u^* associated to the eigenvalue λ_1 of A, where $\|\cdot\|$ denotes the Euclidean norm.

The formulation (4.2.0.1) is inspired by the previous work [102], which considers the minimization of $-1/f(x)$ instead of $f(x)$. These two optimization problems are clearly equivalent and behave very similarly close to the optimum x^*. For a local convergence analysis, as the one performed in [102], the choice between the two optimization problems does not make a significant difference. For attaining results of a more global nature, this choice matters and it turns out that our new formulation (4.2.0.1) is more suitable.

Remark 4.1 *Our work also applies to generalized eigenvalue problems of the form $A - \lambda M$, for SPD matrices A and M. The additional matrix M*

can be absorbed by setting $\widehat{A} = M^{-1/2}AM^{-1/2}$, $\widehat{B} = M^{-1/2}BM^{-1/2}$ *and* $\widehat{x} = \widehat{B}^{-1/2}M^{-1/2}B^{1/2}x$, *and one obtains the same type of optimization problem* (4.2.0.1), *simply with* A, B *and* x *replaced by* $\widehat{A}$, $\widehat{B}$ *and* $\widehat{x}/\|\widehat{x}\|$.

We view $\mathbb{S}^{n-1}$ as a Riemannian submanifold of $\mathbb{R}^n$ with the restricted Euclidean metric (see Section 1.3.1.1). Minimizing (4.2.0.1) by the *Riemannian gradient descent* method yields the recurrence

$$x_{t+1} = \exp_{x_t}(-\eta_t \operatorname{grad} f(x_t)), \tag{4.2.0.2}$$

for an initial vector $x_0 \in \mathbb{S}^{n-1}$, where grad denotes the Riemannian gradient in the sphere and $\exp_{x_t}$ denotes the exponential map at x_t on $\mathbb{S}^{n-1}$ (see Section 1.3.1.1 for explicit formulas). We impose the natural restriction

$$0 < \eta_t < \frac{\pi}{2\|\operatorname{grad} f(x_t)\|} \tag{4.2.0.3}$$

on the step size.

The following proposition shows that the recurrence (4.2.0.2) is a variant of PINVIT (4.1.0.1) that uses a different step size[10] and normalization.

Proposition 4.1 *Consider the iterates* x_t *produced by the recurrence* (4.2.0.2) *with a step size satisfying* (4.2.0.3). *Then the transformed vectors* $u_t := B^{-1/2}x_t$ *satisfy the recurrence*

$$u_{t+1} = \beta_{t+1}(u_t - \eta_t^* B^{-1} r_t), \tag{4.2.0.4}$$

with a certain step size $\eta_t^* > 0$, *a normalization* $\beta_{t+1} > 0$ *chosen such that* $\|u_{t+1}\|_B = 1$, *and the residual* $r_t = Au_t - \lambda(u_t)u_t$.

Proof A direct calculation of the Euclidean gradient of f shows

$$\nabla f(x_t) = -\frac{2(B^{-1}x_t + f(x_t)B^{-1/2}AB^{-1/2}x_t)}{\|A^{1/2}B^{-1/2}x_t\|^2}. \tag{4.2.0.5}$$

Because $I - x_t x_t^T$ is the orthogonal projection to the tangent space of the sphere at x_t, the Riemannian gradient is given by (see, e.g., [3, Example 3.6.1])

$$\operatorname{grad} f(x_t) = (I - x_t x_t^T)\nabla f(x_t) = \nabla f(x_t), \tag{4.2.0.6}$$

where the latter equality follows from $x_t^T \nabla f(x_t) = 0$. In particular, this implies that $\operatorname{grad} f(x_t)$ is zero if and only if the residual r_t is zero. In this case, the recurrence (4.2.0.4) holds trivially. We may therefore assume $\operatorname{grad} f(x_t) \neq 0$ in the following.

[10] Recall that PINVIT can use step size 1 thanks to the normalization of the preconditioner implied by (4.1.0.3).

Using the explicit expression of the exponential map in the sphere, the recurrence (4.2.0.2) is rewritten as

$$\begin{aligned} x_{t+1} &= \cos(\|\eta_t \operatorname{grad} f(x_t)\|)x_t - \sin(\|\eta_t \operatorname{grad} f(x_t)\|)\frac{\operatorname{grad} f(x_t)}{\|\operatorname{grad} f(x_t)\|} \\ &= \beta_{t+1}\big(x_t - \overline{\eta}_t \operatorname{grad} f(x_t)\big), \end{aligned}$$

where we set

$$\beta_{t+1} := \cos(\|\eta_t \operatorname{grad} f(x_t)\|) \quad \text{and} \quad \overline{\eta}_t := \frac{\tan(\|\eta_t \operatorname{grad} f(x_t)\|)}{\|\operatorname{grad} f(x_t)\|}.$$

By (4.2.0.3), $\overline{\eta}_t$ is well defined and positive. Substituting $u_t = B^{-1/2}x_t$ and using (4.2.0.5) and (4.2.0.6), we obtain that

$$\begin{aligned} u_{t+1} &= \beta_{t+1}\Big(u_t + \frac{2\overline{\eta}_t}{\|A^{1/2}B^{-1/2}x_t\|^2}\big(B^{-1}u_t + f(x_t)B^{-1}Au_t\big)\Big) \\ &= \beta_{t+1}\left(u_t + \frac{2\overline{\eta}_t}{u_t^TAu_t}\Big(B^{-1}u_t - \frac{1}{\lambda(u_t)}B^{-1}Au_t\Big)\right) \\ &= \beta_{t+1}(u_t - \eta_t^*B^{-1}r_t), \end{aligned}$$

with

$$\eta_t^* := \frac{2\overline{\eta}_t u_t^Tu_t}{(u_t^TAu_t)^2} = \frac{2\tan(\|\eta_t \operatorname{grad} f(x_t)\|)u_t^Tu_t}{\|\operatorname{grad} f(x_t)\|(u_t^TAu_t)^2} > 0.$$

By definition, x_{t+1} is in the sphere and, therefore, it follows immediately that $\|u_{t+1}\|_B = 1$. ■

4.3 Quality of preconditioner

In this section, we discuss quantities that measure the quality of the preconditioner B in the context of preconditioned eigenvalue solvers.

4.3.1 Global: spectral equivalence

For any SPD matrices A, B, there exist constants $0 < \nu_{\min} \le \nu_{\max}$ such that

$$\nu_{\min}(x^TBx) \le x^TAx \le \nu_{\max}(x^TBx), \quad \forall x, \tag{4.3.1.1}$$

a property sometimes called spectral equivalence. Equivalently,

$$\nu_{\min}\|x\|^2 \le \|A^{1/2}B^{-1/2}x\|^2 \le \nu_{\max}\|x\|^2, \quad \forall x. \tag{4.3.1.2}$$

The tightest bounds are obtained by choosing $\nu_{\min}$ and $\nu_{\max}$ as the smallest and largest eigenvalues of AB^{-1}, respectively. As we will see below, their ratio $\kappa_\nu := \nu_{\max}/\nu_{\min}$ determines the convergence rate of PINVIT and other preconditioned eigenvalue solvers.

While (4.3.1.1) can always be satisfied, ideally κ_ν is not too large. In particular, when A arises from the discretization of a partial differential equation, a good preconditioner B keeps κ_ν bounded as the discretization is refined; see also Section 4.5.

The inequality (4.3.1.1) only implies the condition (4.1.0.3) required by the convergence (analysis) of PINVIT when B is *scaled* in a suitable manner. According to [84], preconditioning with ηB^{-1} instead of B^{-1} with $\eta = 2/(\nu_{\max} + \nu_{\min})$ leads to $\alpha_B = (\kappa_\nu - 1)/(\kappa_\nu + 1) < 1$ in (4.1.0.3).

4.3.2 Local: angle of distortion

Our condition on the initial vector will be based on an *angle of distortion* φ, which measures the distortion induced by the preconditioner at the eigenvector u^*:

$$\varphi := \arcsin \frac{\|u^*\|^2}{\|u^*\|_B \|u^*\|_{B^{-1}}} \in (0, \pi/2]. \tag{4.3.2.1}$$

For the vector $x^* = \frac{B^{1/2}u^*}{\|B^{1/2}u^*\|}$, we have that

$$\frac{x^{*T} B^{-1} x^*}{\|x^*\| \|B^{-1}x^*\|} = \frac{\|u^*\|^2}{\|u^*\|_B \|u^*\|_{B^{-1}}} = \sin\varphi.$$

In other words, φ is complementary to the angle between x^* and $B^{-1}x^*$, as illustrated in 4.1.

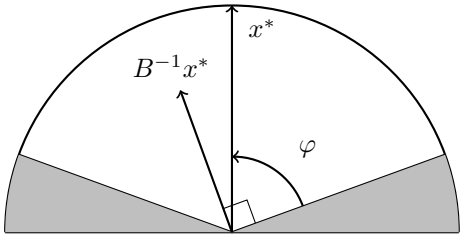


Figure 4.1: Angle of distortion φ. Vectors x in the white region satisfy $\mathrm{dist}(x, x^*) < \varphi$.

For $x \in \mathbb{S}^{n-1}$, we let

$$\mathrm{dist}(x, x^*) := \arccos(x^T x^*)$$

denote the angle between x^* and x, which happens to be the intrinsic Riemannian distance in the sphere. By suitably choosing the sign of x^*, we may always assume that $\mathrm{dist}(x, x^*) \in [0, \pi/2]$. When $\mathrm{dist}(x, x^*) < \varphi$, the following lemma establishes a lower bound on $x^T B^{-1} x^*$ that will play an important role for the so-called weak-quasi convexity property of the function f defined in (4.2.0.1); see Proposition 4.6 below.

Lemma 4.2 *With the notation introduced above, we have that*

$$x^T B^{-1} x^* \geq \frac{\|u^*\|_{B^{-1}}^2}{\|u^*\|^2} \Big(\cos\big(\operatorname{dist}(x, x^*)\big) - \cos\varphi \Big).$$

holds for any $x \in \mathbb{S}^{n-1}$ *with* $\operatorname{dist}(x, x^*) < \varphi$.

Proof Set $\sigma := \|u^*\|_{B^{-1}}^2 / \|u^*\|^2$. By the Cauchy–Schwarz inequality,

$$x^T B^{-1} x^* = \sigma x^T x^* + x^T (B^{-1} - \sigma I) x^* \geq \sigma x^T x^* - \|(B^{-1} - \sigma I) x^*\|.$$

On the other hand, it holds that

$$\frac{\|(B^{-1} - \sigma I) x^*\|^2}{\sigma^2} = \frac{\|B^{-1/2} u^* - \sigma B^{1/2} u^*\|^2}{\sigma^2 \|u^*\|_B^2} = 1 - \frac{\|u^*\|^4}{\|u^*\|_B^2 \|u^*\|_{B^{-1}}^2} = \cos^2\varphi,$$

where the second equality follows by expanding the square. The result follows from combining the two relationships, using that $\cos(\operatorname{dist}(x, x^*)) = x^T x^*$. ■

In the absence of preconditioning, that is, $B = I$, the angle of distortion is $\varphi = \pi/2$ and the inequality of Lemma 4.2 becomes a trivial equality. The same holds when $B \neq I$ in the (unrealistic) scenario that u^* is also an eigenvector of B, which in particular holds when $B = A$. In the general case, one expects that φ is still close to $\pi/2$ or, equivalently, $\cos\varphi \approx 0$ for a good preconditioner B. From (4.3.1.1), one immediately obtains the bound

$$\cos^2\varphi \leq 1 - \kappa_\nu^{-1}. \tag{4.3.2.2}$$

However, this bound is often not sharp and we will establish much tighter bounds for specific preconditioners in Section 4.6.

The following lemma provides a useful variational representation of $\cos\varphi$.

Lemma 4.3 *The angle of distortion* φ *satisfies*

$$\cos\varphi = \sup_{v^T u^* = 0} \frac{v^T B^{-1} u^*}{\|v\|_{B^{-1}} \|u^*\|_{B^{-1}}}. \tag{4.3.2.3}$$

Proof The supremum in (4.3.2.3) is attained by the B^{-1}-orthogonal projection of u^* onto the subspace $\operatorname{span}\{u^*\}^\perp$. This projection is given by the vector

$$v_* = u^* - \frac{\|u^*\|^2}{\|u^*\|_B^2} B u^*, \tag{4.3.2.4}$$

which follows from verifying that $v_*^T u^* = 0$ and $v_*^T B^{-1}(u^* - v_*) = \frac{\|u^*\|^2}{\|u^*\|_B^2} v_*^T u^* = 0$.

Note that u^*, v_* and $u^* - v_*$ form a right triangle with respect to the B^{-1}-inner product, where u^* is the hypotenuse. By Pythagoras,

$$\frac{|v_*^T B^{-1} u^*|^2}{\|v_*\|_{B^{-1}}^2 \|u^*\|_{B^{-1}}^2} = 1 - \frac{|(u^* - v_*)^T B^{-1} u^*|^2}{\|(u^* - v_*)\|_{B^{-1}}^2 \|u^*\|_{B^{-1}}^2} = 1 - \frac{\|u^*\|^4}{\|u^*\|_B^2 \|u^*\|_{B^{-1}}^2} = \cos^2\varphi,$$

where we use the definition (4.3.2.4) of v_* in the second equality, and the definition (4.3.2.1) of φ in the third equality. ■

Remark 4.2 *For the situation considered in this section, the definition of leading angle from [102, Definition 6] amounts to*

$$\vartheta \equiv \vartheta(I_n, -1/\lambda_1; f) = \inf_{f(x)\leq -1/\lambda_1} \inf_{v^T x = 0} \arccos\Big(\frac{|v^T B^{-1} x|}{\|v\|_{B^{-1}} \|x\|_{B^{-1}}}\Big).$$

Similar to the proof of Lemma 4.3*, one can show that*

$$\vartheta = \arcsin \frac{\|u^*\|_B^2}{\|Bu^*\|\|u^*\|}.$$

Comparing with the definition of φ in (4.3.2.1)*, one observes that both ϑ and φ are angles between Bu^* and u^*, one with respect to the standard Euclidean inner product and the other with respect to the B^{-1}-inner product.*

4.4 Convergence analysis

In this section, we study the convergence of the Riemannian gradient descent method (4.2.0.2) or, equivalently, the PINVIT-like method (4.2.0.4). Our analysis utilizes concepts developed in Sections 2 and 3 for analyzing non-preconditioned eigenvalue solvers. In particular, we will use *smoothness* and *weak-quasi-strong convexity* of the objective function f defined in (4.2.0.1) to show that the distance of the iterates (4.2.0.2) to x^* contracts linearly.

4.4.1 Smoothness-type property

Our analysis requires the following smoothness-type property, parametrized by a function $\gamma(x) > 0$:

$$f(x) - f^* \geq \frac{1}{2\gamma(x)} \|\operatorname{grad} f(x)\|^2, \quad \forall x \in \mathbb{S}^{n-1}, \tag{4.4.1.1}$$

where $f^* := f(x^*) = -1/\lambda_1$ denotes the minimum of f. Note that standard smoothness in (convex) optimization implies (4.4.1.1), but not vice versa. This is similar to Proposition 3.5.

Proposition 4.4 *The smoothness-type property* (4.4.1.1) *holds with*

$$\gamma(x) = \frac{\nu_{\max} \cdot (\lambda_1^{-1} - \lambda_n^{-1})}{\|A^{1/2} B^{-1/2} x\|^2}.$$

Proof Using the transformation

$$y(x) := \frac{A^{1/2} B^{-1/2} x}{\|A^{1/2} B^{-1/2} x\|} \in \mathbb{S}^{n-1}, \tag{4.4.1.2}$$

we get

$$f(x) = \overline{f}(y(x)) := -y(x)^T A^{-1} y(x). \tag{4.4.1.3}$$

with $\min\limits_{y \in \mathbb{S}^{n-1}} \overline{f}(y) = f^* = -1/\lambda_1$. Since $\overline{f}$ is the Rayleigh quotient for $-A^{-1}$, we can use the smoothness property of Proposition 3.5:

$$f(x) - f^* = \overline{f}(y(x)) - f^* \geq \frac{1}{2(\lambda_1^{-1} - \lambda_n^{-1})} \|\operatorname{grad} \overline{f}(y(x))\|^2. \tag{4.4.1.4}$$

It remains to phrase this property in terms of x instead of y.

By the chain rule, we have $\mathrm{d}f(x) = \mathrm{d}\overline{f}(y(x))\,\mathrm{d}y(x)$, which implies

$$\operatorname{grad} f(x) = \mathrm{d}y^T(x) \operatorname{grad} \overline{f}(y) \quad \text{and} \quad \|\operatorname{grad} f(x)\| \leq \|\mathrm{d}y(x)\| \|\operatorname{grad} \overline{f}(y)\|. \tag{4.4.1.5}$$

To lower bound the right-hand side of (4.4.1.4), we thus need to upper bound the spectral norm of $\mathrm{d}y(x)$. Denote $C := A^{1/2} B^{-1/2}$. A direct calculation shows that

$$\mathrm{d}y(x)v = \frac{Cv}{\|Cx\|} - Cx \frac{x^T C^T C v}{\|Cx\|^3} = \frac{1}{\|Cx\|} \left(I - \frac{Cxx^T C^T}{\|Cx\|^2} \right) Cv$$

holds for any v. Taking the Euclidean norm and noticing that the matrix in parentheses is an orthogonal projector, we obtain

$$\|\mathrm{d}y(x)v\| \leq \frac{\|Cv\|}{\|Cx\|} \leq \frac{\sqrt{\nu_{\max}} \|v\|}{\|A^{1/2} B^{-1/2} x\|}$$

and, hence, $\|\mathrm{d}y(x)\| \leq \sqrt{\nu_{\max}} / \|A^{1/2} B^{-1/2} x\|$. Plugging this inequality into (4.4.1.5), we get

$$\|\operatorname{grad} \overline{f}(y(x))\|^2 \geq \frac{\|A^{1/2} B^{-1/2} x\|^2}{\nu_{\max}} \|\operatorname{grad} f(x)\|^2.$$

Together with the bound (4.4.1.4), this gives the desired inequality:

$$f(x) - f^* \geq \frac{\|A^{1/2} B^{-1/2} x\|^2}{2\nu_{\max}(\lambda_1^{-1} - \lambda_n^{-1})} \|\operatorname{grad} f(x)\|^2.$$

■

It is worth noting that Proposition 4.5 combined with (4.3.1.2) give the global bound

$$\gamma(x) \leq \kappa_\nu \cdot (\lambda_1^{-1} - \lambda_n^{-1}), \quad \forall x \in \mathbb{S}^{n-1}. \tag{4.4.1.6}$$

4.4.2 Quadratic growth

In this and the next section, we derive two properties of f that correspond to weakened notions of strong convexity. We recall that $\operatorname{dist}(x_1, x_2)$ denotes the

angle between two vectors x_1, x_2. If $\|x_1\| = \|x_2\| = 1$, it follows from a simple geometrical argument that

$$\|x_1 - x_2\| \leq \operatorname{dist}(x_1, x_2) \leq \frac{\pi}{2}\|x_1 - x_2\|. \tag{4.4.2.1}$$

The next property is an analogue of Proposition 3.2.

Proposition 4.5 *The function f satisfies*

$$f(x) - f^* \geq \frac{\mu(x)}{2} \operatorname{dist}^2(x, x^*), \quad \forall x \in \mathbb{S}^{n-1},$$

with

$$\mu(x) := \frac{8\nu_{\min} \cdot (\lambda_1^{-1} - \lambda_2^{-1})\|u^*\|_B}{\pi^2 \|A^{1/2}B^{-1/2}x\| \|u^*\|_A}.$$

Proof As in the proof of Proposition 4.4, we apply the transformation $y(x)$ from (4.4.1.2) to obtain the transformed objective function $\overline{f}$ in (4.4.1.3). By the quadratic growth of $\overline{f}$ of Proposition 3.2, we have

$$f(x) - f^* = \overline{f}(y(x)) - f^* \geq (\lambda_1^{-1} - \lambda_2^{-1}) \operatorname{dist}^2(y(x), u^*). \tag{4.4.2.2}$$

It thus remains to lower bound $\operatorname{dist}(y(x), u^*)$ in terms of $\operatorname{dist}(x, x^*)$. For this purpose, we may assume $\|u^*\| = 1$ without loss of generality.

We first rewrite

$$y(x) - u^* = A^{1/2}B^{-1/2}z \quad \text{with} \quad z = \frac{x}{\|A^{1/2}B^{-1/2}x\|} - B^{1/2}A^{-1/2}u^*.$$

Using (4.4.2.1), we obtain that

$$\operatorname{dist}^2(y(x), u^*) \geq \|y(x) - u^*\|^2 = \|A^{1/2}B^{-1/2}z\|^2 \geq \nu_{\min}\|z\|^2. \tag{4.4.2.3}$$

Since $x^* = B^{1/2}u^*/\|u^*\|_B$ and $A^{-1/2}u^* = \lambda_1^{-1/2}u^*$, we can also write

$$z = \frac{x}{\|A^{1/2}B^{-1/2}x\|} - \frac{\|u^*\|_B}{\|u^*\|_A}x^*,$$

For any α_1, $\alpha_2 \in \mathbb{R}$ and x_1, $x_2 \in \mathbb{S}^{n-1}$, it holds that

$$\|\alpha_1 x_1 - \alpha_2 x_2\|^2 = \alpha_1^2 + \alpha_2^2 - 2\alpha_1\alpha_2 x_1^T x_2 \geq \alpha_1\alpha_2\|x_1 - x_2\|^2.$$

Using (4.4.2.1) once more, we can therefore bound

$$\|z\|^2 \geq \frac{\|u^*\|_B}{\|A^{1/2}B^{-1/2}x\|\|u^*\|_A}\|x - x^*\|^2 \geq \frac{4\|u^*\|_B}{\pi^2\|A^{1/2}B^{-1/2}x\|\|u^*\|_A} \operatorname{dist}^2(x, x^*).$$

Combined with (4.4.2.2) and (4.4.2.3), it yields the inequality

$$f(x) - f^* \geq \frac{4\nu_{\min}(\lambda_1^{-1} - \lambda_2^{-1})\|u^*\|_B}{\pi^2\|A^{1/2}B^{-1/2}x\|\|u^*\|_A} \operatorname{dist}^2(x, x^*),$$

which is the desired result. ■

By (4.3.1.2), the quantity $\mu(x)$ of Proposition 4.5 admits the constant lower bound

$$\mu(x) \geq \frac{8(\lambda_1^{-1} - \lambda_2^{-1})}{\pi^2 \kappa_\nu} =: \mu_0, \quad \forall x \in \mathbb{S}^{n-1}. \tag{4.4.2.4}$$

This shows that μ_0-strong convexity implies the quadratic growth established by Proposition 4.5, with $\mu(x)$ replaced by the constant μ_0. This constant features the key quantities in the classical convergence result (4.1.0.2): the spectral gap of A^{-1} measured by $\lambda_1^{-1} - \lambda_2^{-1}$ and the spectral equivalence (4.3.1.1) of the preconditioner measured by κ_ν.

4.4.3 Weak-quasi convexity

We now establish our second convexity-like property that is essential for the analysis of the Riemannian gradient descent method (4.2.0.2). This is an analogue of Proposition 3.1.

Proposition 4.6 *Suppose that $x \in \mathbb{S}^{n-1}$ satisfies $\operatorname{dist}(x, x^*) < \varphi$ with the angle of distortion φ defined in* (4.3.2.1). *Then*

$$\langle \operatorname{grad} f(x), -\operatorname{Log}_x(x^*) \rangle \geq 2a(x)\big(f(x) - f(x^*)\big), \tag{4.4.3.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product, and

$$a(x) := \frac{\lambda_1 \|u^*\|_{B^{-1}}^2 (\cos(\operatorname{dist}(x, x^*)) - \cos\varphi)}{\|A^{1/2} B^{-1/2} x\|^2 \|u^*\|^2}.$$

Proof To simplify notation, we set $\theta_x := \operatorname{dist}(x, x^*)$. Because $\|P_x x^*\|^2 = 1 - (x^T x^*)^2 = 1 - \cos^2\theta_x$, we can write the Riemannian logarithm from (1.3.1.1) as

$$\operatorname{Log}_x(x^*) = \frac{\theta_x}{\sin\theta_x} P_x x^*.$$

As mentioned in the proof of 4.1, $\operatorname{grad} f(x) = P_x \nabla f(x) = \nabla f(x)$. We therefore get

$$\langle \operatorname{grad} f(x), -\operatorname{Log}_x(x^*) \rangle = -\frac{\theta_x}{\sin\theta_x} \langle P_x \nabla f(x), P_x x^* \rangle = -\frac{\theta_x}{\sin\theta_x} \langle \nabla f(x), x^* \rangle.$$

Using the expression (4.2.0.5) for $\nabla f(x)$, $B^{-1/2} A B^{-1/2} x^* = \lambda_1 B^{-1} x^*$, and $f^* = -1/\lambda_1$, one gets

$$\langle \nabla f(x), x^* \rangle = -\frac{2\lambda_1 x^T B^{-1} x^*}{\|A^{1/2} B^{-1/2} x\|^2} \big(f(x) - f^*\big).$$

Combining the two equations above gives

$$\langle \operatorname{grad} f(x), -\operatorname{Log}_x(x^*)\rangle = \frac{2\lambda_1\theta_x(x^TB^{-1}x^*)}{\|A^{1/2}B^{-1/2}x\|^2\sin\theta_x}\big(f(x)-f^*\big).$$

The result now follows from the bound on $x^TB^{-1}x^*$ established in 4.2, additionally using that $\theta_x/\sin\theta_x \geq 1$. ■

Remark 4.3 *If* $\cos(\operatorname{dist}(x,x^*)) \geq \cos\varphi + c\sin^2\varphi$ *for some* $0 < c < 1/2$*, the factor* $a(x)$ *of Proposition* 4.6 *can be bounded by a constant:*

$$a(x) \geq \frac{c\|u^*\|_A^2}{\|A^{1/2}B^{-1/2}x\|^2\|u^*\|_B^2} \geq \frac{c}{\kappa_\nu},$$

This follows from (4.3.1.2), (4.3.2.1)*, and* $Au^* = \lambda_1 u^*$.

4.4.4 Weak-quasi-strong convexity

The quadratic growth and weak-quasi convexity properties established above result in a WQSC property, similarly to Proposition 3.3.

Proposition 4.7 *The function* f *defined in* (4.2.0.1) *satisfies*

$$f(x) - f^* \leq \frac{1}{a(x)}\langle \operatorname{grad} f(x), -\log_x(x^*)\rangle - \frac{\mu(x)}{2}\operatorname{dist}^2(x,x^*),$$

for every $x \in \mathbb{S}^{n-1}$ *satisfying* $\operatorname{dist}(x,x^*) < \varphi$*, with* $\mu(x)$ *and* $a(x)$ *defined in Propositions* 4.5 *and* 4.6*, respectively.*

Proof By Propositions 4.5 and 4.6, we have

$$\frac{\mu(x)}{2}\operatorname{dist}^2(x,x^*) \leq f(x) - f^* \leq \frac{1}{2a(x)}\langle \operatorname{grad} f(x), -\log_x(x^*)\rangle.$$

Note that $\operatorname{dist}(x,x^*) < \varphi$ implies $a(x) > 0$. Applying this inequality twice shows the desired result:

$$\begin{aligned} f(x) - f^* &\leq \frac{1}{2a(x)}\langle \operatorname{grad} f(x), -\log_x(x^*)\rangle + \frac{\mu(x)}{2}\operatorname{dist}^2(x,x^*) - \frac{\mu(x)}{2}\operatorname{dist}^2(x,x^*) \\ &\leq \frac{1}{a(x)}\langle \operatorname{grad} f(x), -\log_x(x^*)\rangle - \frac{\mu(x)}{2}\operatorname{dist}^2(x,x^*). \end{aligned}$$

■

4.4.5 Convergence analysis

Theorem 4.8 below contains the main theoretical result of this section, on the contraction of the error (measured in terms of the angles $\operatorname{dist}(x_t, x^*)$) for the iterates produced by the Riemannian gradient descent method for the problem (4.2.0.1). The condition on the initial vector prominently features the angle of distortion φ defined in (4.3.2.1), whereas the contraction rate involves the relative spectral gap for A^{-1} and the quantity $\kappa_\nu = \nu_{\max}/\nu_{\min}$ measuring the spectral equivalence (4.3.1.1) of the preconditioner.

Theorem 4.8 *For an eigenvector u^* associated with the smallest eigenvalue λ_1 and an SPD preconditioner B, let $x^* := B^{1/2}u^*/\|B^{1/2}u^*\|$. Apply the Riemannian gradient descent method (4.2.0.2) to the optimization problem (4.2.0.1), with a starting vector $x_0 \in \mathbb{S}^{n-1}$ such that*

$$\operatorname{dist}(x_0, x^*) < \varphi, \tag{4.4.5.1}$$

and a step size η_t satisfying

$$\eta_t \leq \frac{a(x_t)}{\gamma(x_t)} = \frac{\lambda_1 \|u^*\|_{B^{-1}}^2 (\cos(\operatorname{dist}(x_t, x^*)) - \cos\varphi)}{\nu_{\max}\|u^*\|^2(\lambda_1^{-1} - \lambda_n^{-1})},$$

with $\gamma(x)$ and $a(x)$ defined in Propositions 4.4 and 4.6. Then the iterates x_t produced by Algorithm (4.2.0.2) satisfy

$$\operatorname{dist}^2(x_{t+1}, x^*) \leq (1 - \xi_t)\operatorname{dist}^2(x_t, x^*),$$

where $\xi_t := \eta_t \mu(x_t) a(x_t)$ with $\mu(x)$ defined in Proposition 4.5, respectively. When fixing the step size $\eta_t = a(x_t)/\gamma(x_t)$ we have

$$\xi_t = \frac{8\lambda_1^2 \|u^*\|_B \|u^*\|_{B^{-1}}^4}{\pi^2 \|u^*\|^4 \|u^*\|_A} \frac{(\cos(\operatorname{dist}(x_t, x^*)) - \cos\varphi)^2}{\|A^{1/2}B^{-1/2}x_t\|^3} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})} \tag{4.4.5.2}$$

bounded below by a positive constant, and $\operatorname{dist}(x_t, x^*)$ converges linearly to zero.

Proof By the structure of (4.2.0.2), we have $\operatorname{Log}_{x_t}(x_{t+1}) = -\eta_t \operatorname{grad} f(x_t)$. Since $\operatorname{dist}(x, y) = \|\operatorname{Log}_x(y)\|$, it follows by Proposition 1.15 that

$$\begin{aligned} \operatorname{dist}^2(x_{t+1}, x^*) &\leq \|-\eta_t \operatorname{grad} f(x_t) - \operatorname{Log}_{x_t}(x^*)\|^2 \\ &= \eta_t^2 \|\operatorname{grad} f(x_t)\|^2 + \operatorname{dist}^2(x_t, x^*) + 2\eta_t \langle \operatorname{grad} f(x_t), \operatorname{Log}_{x_t}(x^*)\rangle. \end{aligned} \tag{4.4.5.3}$$

By Propositions 4.7 and 4.4, we have

$$\begin{aligned} \frac{1}{a(x_t)}\langle \operatorname{grad} f(x_t), \operatorname{Log}_{x_t}(x^*)\rangle &\leq f^* - f(x_t) - \frac{\mu(x_t)}{2}\operatorname{dist}^2(x_t, x^*) \\ &\leq -\frac{1}{2\gamma(x_t)}\|\operatorname{grad} f(x_t)\|^2 - \frac{\mu(x_t)}{2}\operatorname{dist}^2(x_t, x^*). \end{aligned}$$

Multiplying with $2\eta_t a(x_t)$ and using the hypothesis $\eta_t \leq a(x_t)/\gamma(x_t)$, this gives

$$\begin{aligned} 2\eta_t \langle \operatorname{grad} f(x_t), \operatorname{Log}_{x_t}(x^*) \rangle &\leq -\frac{\eta_t a(x_t)}{\gamma(x_t)} \|\operatorname{grad} f(x_t)\|^2 - \eta_t \mu(x_t) a(x_t) \operatorname{dist}^2(x_t, x^*) \\ &\leq -\eta_t^2 \|\operatorname{grad} f(x_t)\|^2 - \eta_t \mu(x_t) a(x_t) \operatorname{dist}^2(x_t, x^*). \end{aligned}$$

Plugging this inequality into (4.4.5.3) proves the first part of the theorem:

$$\operatorname{dist}^2(x_{t+1}, x^*) \leq (1 - \eta_t \mu(x_t) a(x_t)) \operatorname{dist}^2(x_t, x^*).$$

The expression (4.4.5.2) directly follows from the definitions of $a(x_t), \gamma(x_t), \mu(x_t)$, implying $\operatorname{dist}(x_t, x^*) \leq \operatorname{dist}(x_0, x^*)$. Finally, the claimed linear convergence can be concluded from the fact that ξ_t admits the constant lower bound

$$\xi_t \geq \frac{8\lambda_1^2 \|u^*\|_B \|u^*\|_{B^{-1}}^4}{\pi^2 \|u^*\|^4 \|u^*\|_A} \frac{(\cos(\operatorname{dist}(x_0, x^*)) - \cos\varphi)^2}{\nu_{\max}^{3/2}} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})} > 0,$$

where we use $\operatorname{dist}(x_t, x^*) \leq \operatorname{dist}(x_0, x^*)$ and the spectral equivalence (4.3.1.2). ■

Propositions 4.1, 4.8 establish an error contraction, with contraction rate $1 - \xi_t$, also for the PINVIT-like method (4.2.0.4), *if* the step size restriction (4.2.0.3) is satisfied. Using the smoothness-type property (4.4.1.1) and the weak-quasi convexity property (4.4.3.1), it follows that

$$\frac{a(x_t)}{\gamma(x_t)} \leq \frac{2a(x_t)\big(f(x_t) - f(x^*)\big)}{\|\operatorname{grad} f(x_t)\|^2} \leq \frac{-\langle \operatorname{grad} f(x), \operatorname{Log}_{x_t}(x^*) \rangle}{\|\operatorname{grad} f(x_t)\|^2} < \frac{\pi}{2\|\operatorname{grad} f(x_t)\|},$$

where the last inequality uses that $\|\operatorname{Log}_{x_t}(x^*)\| = \operatorname{dist}(x_t, x^*) < \pi/2$ is implied by (4.4.5.1). Hence, the step size restriction $\eta_t \leq a(x_t)/\gamma(x_t)$ imposed by Proposition 4.8 always implies (4.2.0.3). In terms of the PINVIT iterates $u_t = B^{-1/2} x_t$, the initial condition (4.4.5.1) takes the form

$$\operatorname{dist}(x_0, x^*) = \operatorname{dist}_B(u_0, u^*) := \arccos\Big(\frac{u_0^T B u^*}{\|u_0\|_B \|u^*\|_B}\Big) < \varphi, \tag{4.4.5.4}$$

where the sign of u^* is chosen such that $u_0^T B u^* \geq 0$.

The following corollary establishes convergence for a constant step size.

Corollary 4.9 *If*

$$\cos\big(\operatorname{dist}(x_0, x^*)\big) \geq \cos\varphi + c\sin^2\varphi \;\; \textit{and} \;\; \eta = \frac{c}{\kappa_\nu^2(\lambda_1^{-1} - \lambda_n^{-1})} \tag{4.4.5.5}$$

for some $0 < c < 1/2$, then the Riemannian gradient descent method (4.2.0.2) *with step size η produces iterates x_t satisfying*

$$\operatorname{dist}^2(x_t, x^*) \leq \left(1 - \frac{8c^2(\lambda_1^{-1} - \lambda_2^{-1})}{\pi^2 \kappa_\nu^4 (\lambda_1^{-1} - \lambda_n^{-1})}\right)^t \operatorname{dist}^2(x_0, x^*). \tag{4.4.5.6}$$

Thus, x_t converges linearly to x^*.

Proof The proof proceeds by induction on t. The result for $t = 0$ is trivial. Suppose (4.4.5.6) holds for some $t \geq 1$ and we now show that it also holds for $t + 1$. From (4.4.5.6) and (4.4.5.5), it follows that

$$\cos\big(\operatorname{dist}(x_t, x^*)\big) \geq \cos\big(\operatorname{dist}(x_0, x^*)\big) \geq \cos\varphi + c\sin^2\varphi.$$

As shown in the bound (4.4.1.6) and Remark 4.3, we have $\gamma(x_t) \leq \kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})$ and $a(x_t) \geq c/\kappa_\nu$. Hence, the choice of η in (4.4.5.5) satisfies the condition $\eta \leq a(x_t)/\gamma(x_t)$. By Theorem 4.8, we have

$$\operatorname{dist}^2(x_{t+1}, x^*) \leq (1 - \eta\mu(x_t)a(x_t))\operatorname{dist}^2(x_t, x^*).$$

Using the lower bound (4.4.2.4) on $\mu(x_t)$ and, once again, $a(x_t) \geq c/\kappa_\nu$, the contraction rate can be bounded by

$$1 - \eta\mu(x_t)a(x_t) \leq 1 - \eta\frac{8c(\lambda_1^{-1} - \lambda_2^{-1})}{\pi^2\kappa_\nu^2} = 1 - \frac{8c^2(\lambda_1^{-1} - \lambda_2^{-1})}{\pi^2\kappa_\nu^4(\lambda_1^{-1} - \lambda_n^{-1})}.$$

This completes the induction step. ■

Corollary 4.9 immediately yields a statement on the iteration complexity.

Corollary 4.10 *Suppose that Riemannian gradient descent* (4.2.0.2) *is applied to the function f in* (4.2.0.1) *with starting vector x_0 and step size η satisfying* (4.4.5.5). *Then an approximation x_T of x^* such that* $\operatorname{dist}(x_T, x^*) \leq \epsilon$ *is returned after*

$$T = \mathcal{O}\left(\frac{\kappa_\nu^4}{c^2}\frac{\lambda_1^{-1} - \lambda_n^{-1}}{\lambda_1^{-1} - \lambda_2^{-1}}\log\frac{\operatorname{dist}(x_0, x^*)}{\epsilon}\right)$$

iterations.

The following lemma simplifies the condition on the starting vector in 4.9, at the expense of making it potentially (much) stricter.

Lemma 4.11 *If*

$$\cos^2(\operatorname{dist}(x_0, x^*)) \geq 1 - \frac{1 - 2c}{\kappa_\nu}, \quad 0 < c < 1/2,$$

then the condition (4.4.5.5) *on the starting vector x_0 is satisfied.*

Proof To establish the result, we show that $1 - \frac{1-2c}{\kappa_\nu} \geq (\cos\varphi + c\sin^2\varphi)^2$ holds for every $0 < c < 1/2$. For this purpose, consider the quadratic function

$$\begin{aligned} q(c) &= \big(\cos\varphi + c\sin^2\varphi\big)^2 - 1 + (1 - 2c)/\kappa_\nu \\ &= (\sin^4\varphi)c^2 + 2\big(\cos\varphi\sin^2\varphi - 1/\kappa_\nu\big)c + 1/\kappa_\nu - \sin^2\varphi. \end{aligned}$$

By the bound (4.3.2.2), we know that $q(0) = 1/\kappa_\nu - \sin^2\varphi \leq 0$. At the same time, we have

$$q(1/2) = \frac{1}{4}\sin^4\varphi + \cos\varphi\sin^2\varphi - \sin^2\varphi \leq 0.$$

Because q is quadratic with leading non-negative coefficient, it follows that $q(c) \leq 0$ for every $0 < c < 1/2$, which completes the proof. ∎

We now derive the *asymptotic* convergence rate implied by Theorem 4.8. This asymptotic rate is much more favorable than the non-asymptotic rate established in Corollary 4.9.

Proposition 4.12 *For the Riemannian gradient descent method* (4.2.0.2) *with step size* $\eta_t = a(x_t)/\gamma(x_t)$*, the quantity* ξ_t *determining the convergence rate* $1 - \xi_t$*, according to Theorem* 4.8*, satisfies*

$$\xi_\infty := \lim_{t\to\infty} \xi_t = \frac{8}{\pi^2(1+\cos\varphi)^2} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})}.$$

Proof Theorem 4.8 shows that $\operatorname{dist}(x_t, x^*) \to 0$ as $t \to \infty$. Inserted into (4.4.5.2), this gives

$$\xi_\infty = \frac{8\lambda_1^2\|u^*\|_B\|u^*\|_{B^{-1}}^4}{\pi^2\|u^*\|^4\|u^*\|_A} \frac{(1-\cos\varphi)^2}{\|A^{1/2}B^{-1/2}x^*\|^3} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})}.$$

Using the relations

$$\lambda_1^2 = \frac{\|u^*\|_A^4}{\|u^*\|^4}, \quad \|A^{1/2}B^{-1/2}x^*\|^3 = \frac{\|u^*\|_A^3}{\|u^*\|_B^3} \quad \text{and} \quad \sin\varphi = \frac{\|u^*\|^2}{\|u^*\|_B\|u^*\|_{B^{-1}}},$$

the expression for ξ_∞ simplifies to

$$\xi_\infty = \frac{8(1-\cos\varphi)^2}{\pi^2\sin^4\varphi} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})} = \frac{8}{\pi^2(1+\cos\varphi)^2} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\kappa_\nu(\lambda_1^{-1} - \lambda_n^{-1})}.$$

∎

The convergence result (4.1.0.2) by Knyazev and Neymeyr shows that the *eigenvalue* approximations of PINVIT converge linearly with the asymptotic convergence rate α^2. When B is optimally scaled, then

$$\alpha = 1 - \frac{2}{\kappa_\nu + 1} \frac{\lambda_1^{-1} - \lambda_2^{-1}}{\lambda_1^{-1}};$$

see Section 4.3.1. On the other hand, Proposition 4.12 establishes the asymptotic convergence rate $1 - \xi_\infty$ for the *eigenvector* approximation error. As the

eigenvalue approximation error is quadratic in the eigenvector approximation error (see, for example, [110, Eq. (27.3)]), it is reasonable to compare ξ_∞ with $1-\alpha$:

$$\xi_\infty = (1-\alpha)\cdot\frac{4}{\pi^2(1+\cos\varphi)^2}\cdot\frac{\kappa_\nu+1}{\kappa_\nu}\cdot(1+\lambda_n^{-1}).$$

Because $0 \le \cos\varphi < 1$, this shows that our asymptotic rate matches (up to a small constant) the sharp rate by Knyazev and Neymeyr.

4.5 Distortion angle for specific preconditioners

The convergence results of the previous section, most notably Theorem 4.8, requires the condition (4.4.5.4) on the initial vector u_0, which can be restated as

$$\frac{u_0^T B u^*}{\|u_0\|_B\|u^*\|_B} > \cos\varphi = \sup_{v^T u^*=0}\frac{v^T B^{-1}u^*}{\|v\|_{B^{-1}}\|u^*\|_{B^{-1}}}, \tag{4.5.0.1}$$

where u^* is an eigenvector belonging to the smallest eigenvalue λ_1 of A. For a (Gaussian) random vector u_0, the left-hand side of (4.5.0.1) is nonzero almost surely, but it is unlikely to be far away from zero. Therefore, a good global convergence guarantee requires $\cos\varphi$ to be small. In this section, we will demonstrate for two specific types of preconditioners that $\cos\varphi$ can be close to zero under reasonable assumptions.

4.5.1 Additive Schwarz preconditioners

Domain decomposition methods (DDM) are widely used strategies for solving large-scale partial differential equations (PDEs). They are based on splitting a PDE, or an approximation of it, into coupled problems on smaller subdomains that collectively form a (possibly overlapping) partition of the original computational domain. A powerful way to analyze and develop DDM is through a subspace perspective [114] that divides the solution space into smaller subspaces, typically corresponding to the geometric structure of the subdomain partition. Here, we consider an additive Schwarz preconditioner as a representative DDM approach. Further details on DDM can be found in several classical references on the topic, such as [109]. The following discussion builds on the previous work [102].

We first briefly describe a relatively standard mathematical setting for elliptic PDEs. On a convex polygonal domain $\Omega \subset \mathbb{R}^d$ with $d = 2$ or 3, consider a symmetric and uniformly positive definite coefficient matrix $\{a_{ij}(x)\}_{i,j=1}^d$ such that $a_{ij}(x) \in C^{0,1}(\overline{\Omega})$ for $i,j = 1,\dots,d$. Let $V_H \subset V_h \subset H_0^1(\Omega)$ be continuous, piecewise linear finite element spaces based on quasi-uniform triangular partitions $\mathcal{T}_H$ and $\mathcal{T}_h$ of Ω, such that $\mathcal{T}_h$ is a refinement of $\mathcal{T}_H$, and $0 < h < H < 1$ are the maximum mesh sizes of $\mathcal{T}_h$ and $\mathcal{T}_H$, respectively. Then the elliptic PDE

eigenvalue problem discretized on V_h takes the following form:

$$\mathcal{A}(u^*, v) = \lambda_1 \langle u^*, v \rangle_2 \quad \forall\, v \in V_h, \quad \text{where } \|u^*\|_2 = 1 \text{ and } u^* \in V_h. \tag{4.5.1.1}$$

Here $\langle \cdot, \cdot \rangle_2$ and $\|\cdot\|_2$ denote the L^2 inner product and norm, respectively, and

$$\mathcal{A}(u, v) := \sum_{i,j=1}^{d} \int_\Omega a_{ij}(x) \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_j} \, \mathrm{d}x. \tag{4.5.1.2}$$

The global solver is the linear operator $A^{-1} : V_h \to V_h$ such that $u \mapsto A^{-1}u$ satisfies

$$\mathcal{A}(A^{-1}u, v) = \langle u, v \rangle_2 \quad \forall\, v \in V_h.$$

We are interested in an additive Schwarz preconditioner B^{-1} for A^{-1}.

To aid in understanding, we present a specific example of additive Schwarz preconditioners, following the structure outlined in [23, Section 7.4].

Example 4.1 (Two-level overlapping domain decomposition preconditioner) *Consider the region $\Omega = [0, 1]^2$. Let $\mathcal{T}_H$ be a coarse triangulation as shown in Figure 4.2. The region Ω is divided into non-overlapping subdomains $\tilde{\Omega}_j$ for $1 \le j \le 16$, which are aligned with $\mathcal{T}_H$. Subsequently, $\mathcal{T}_H$ is further subdivided to obtain the finer triangulation $\mathcal{T}_h$. Define $\Omega_j = \tilde{\Omega}_{j,\delta} \cap \overline{\Omega}$, where $\tilde{\Omega}_{j,\delta}$ is an open set obtained by enlarging $\tilde{\Omega}_j$ by a band of width δ, ensuring Ω_j is aligned with $\mathcal{T}_h$ as shown in Figure 4.2. One often assumes that the overlapping ratio δ/H is bounded below by a constant, which is 0.5 in this case.*

Let $V_j \subset V_h$ denote the subspace of continuous, piecewise linear functions supported in Ω_j for $1 \le j \le 16$. Define the coarse/local solvers A_H^{-1} and A_j^{-1} through

$$\mathcal{A}(A_H^{-1}u_H, v_H) = \langle u_H, v_H \rangle_2 \quad \forall\, v_H \in V_H,$$
$$\mathcal{A}(A_j^{-1}u_j, v_j) = \langle u_j, v_j \rangle_2 \quad \forall\, v_j \in V_j.$$

Then the two-level overlapping domain decomposition preconditioner is given by

$$B^{-1} = I_H A_H^{-1} I_H^T + \sum_{j=1}^{16} I_j A_j^{-1} I_j^T,$$

where $I_H : V_H \mapsto V_h$ and $I_j : V_j \mapsto V_h$ are the natural injection operators, i.e., $I_H v_H = v_H$ for all $v_H \in V_H$, and $I_j v_j = v_j$ for all $1 \le j \le 16$ and $v_j \in V_j$.

Under reasonable assumptions, such as those stated in [109, Assumptions 2.2—2.4], it holds that $\cos\varphi = \mathcal{O}(H)$ as $H \to 0$. To see this, we employ the following results from [102, Lemmas. 34 and 35], which hold under such assumptions:

$$\|B^{-1}u^* - \lambda_H^{-1}u^*\|_{\mathcal{A}} \le c_d \lambda_1^{-1/2} H \quad \text{and} \quad \|v\|_{A^{-1}} \le c_d \|v\|_{B^{-1}} \quad \forall\, v \in V_h, \tag{4.5.1.3}$$

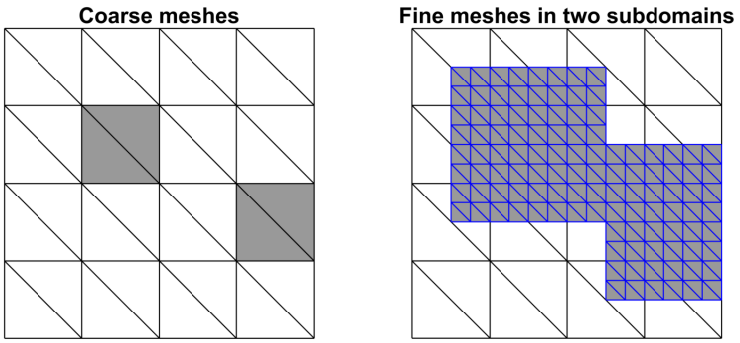


Figure 4.2: Construction of an overlapping domain decomposition. Example and figure taken from [102, Example 30].

where λ_H is the smallest eigenvalue of $\mathcal{A}(\cdot,\cdot)$ in V_H, $\|\cdot\|_{\mathcal{A}}$, $\|\cdot\|_{A^{-1}}$, and $\|\cdot\|_{B^{-1}}$ are the norms induced by $\mathcal{A}(\cdot,\cdot)$, A^{-1}, and B^{-1}, respectively, and $c_d > 0$ is a constant independent of the mesh sizes h, H. For any $v \in V_h$ satisfying $\langle u^*, v\rangle_2 = 0$ and $\|v\|_{B^{-1}} = 1$, the Cauchy–Schwarz inequality yields

$$\langle B^{-1}u^*, v\rangle_2 = \langle B^{-1}u^* - \lambda_H^{-1}u^*, v\rangle_2 \leq \|v\|_{A^{-1}}\|B^{-1}u^* - \lambda_H^{-1}u^*\|_{\mathcal{A}} \leq c_d^2\lambda_1^{-1/2}H.$$

By the variational representation (4.5.0.1) of φ,

$$\cos\varphi = \sup_{\langle u^*,v\rangle_2=0} \frac{\langle B^{-1}u^*, v\rangle_2}{\|v\|_{B^{-1}}\|u^*\|_{B^{-1}}} \leq \frac{c_d^2\lambda_1^{-1/2}H}{\|u^*\|_{B^{-1}}} \leq c_d^3 H. \tag{4.5.1.4}$$

As c_d is independent of h, H, it follows that $\cos\varphi = \mathcal{O}(H)$ as $H \to 0$.

4.5.2 Mixed-precision preconditioners

In this section, we study the condition (4.5.0.1) when using mixed-precision preconditioners as proposed in [64]. For this purpose, we consider two levels of precision: a working precision and a lower precision, for example, IEEE double and single precision. The preconditioner is constructed in lower precision while the rest of the computations are carried out in working precision. For simplicity, the effects of round-off errors in working precision are ignored.

Consider the Cholesky factorization $A = LL^T$, and let $\widehat{L}$ be the Cholesky factor computed in lower precision. We define the preconditioner B as $B^{-1}x := \widehat{L}^{-T}(\widehat{L}^{-1}x)$, which is implemented by solving two triangular linear systems by performing forward and backward substitution in lower precision. By [64, Lemma 3], B^{-1} is a high-quality preconditioner for A, which satisfies

$$\|I - A^{1/2}B^{-1}A^{1/2}\| \leq \frac{\epsilon_l}{1-\epsilon_l}$$

where we assume $\epsilon_l := 4n(3n+1)(\lambda_n/\lambda_1)\mathbf{u}_l < 1$ and $\mathbf{u}_l$ denotes unit roundoff in lower precision. Note that

$$1 - \|I - A^{1/2}B^{-1}A^{1/2}\| \leq \lambda_{\min}(B^{-1}A) \leq \lambda_{\max}(B^{-1}A) \leq 1 + \|I - A^{1/2}B^{-1}A^{1/2}\|.$$

Using the bound (4.3.2.2) for $\cos\varphi$, it follows that

$$\cos^2\varphi \leq 1 - \kappa_\nu^{-1} = 1 - \frac{\lambda_{\min}(B^{-1}A)}{\lambda_{\max}(B^{-1}A)} \leq 1 - \frac{1 - \frac{\epsilon_l}{1-\epsilon_l}}{1 + \frac{\epsilon_l}{1-\epsilon_l}} = 2\epsilon_l.$$

Usually $\epsilon_l \ll 1$ and, hence, $\cos\varphi \leq \sqrt{2\epsilon_l}$ is close to zero. This implies that a random starting vector nearly always satisfies the condition (4.5.0.1). In contrast, the condition $\lambda(u_0) < \lambda_2$ required by the classical analysis of PINVIT does not enjoy any benefit from such a high-quality preconditioner.

4.6 Numerical experiments

In this section, we present some numerical experiments to provide insight into the behavior of φ and a comparison between our initial condition (4.4.5.1) and the classical condition $\lambda(u_0) \in [\lambda_1, \lambda_2)$. All numerical experiments in this section have been implemented in Matlab 2022b and were carried out on an AMD Ryzen 9 6900HX Processor (8 cores, 3.3–4.9 GHz) and 32 GB of RAM.

4.6.1 Laplace eigenvalue problems

The experiments in this section target the smallest eigenvalue for the Laplacian eigenvalue problem with zero Dirichlet boundary condition on the unit square $\Omega = [0,1]^2$:

$$\begin{aligned} -\Delta u &= \lambda u \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{4.6.1.1}$$

We will consider two different scenarios:

AGMG Five-points finite difference discretization of (4.6.1.1) on a regular grid of grid size h, together with an AGMG preconditioner,

DDM Piecewise linear finite element discretization of (4.6.1.1) on a regular mesh of mesh width h, as shown in 4.1, together with a DDM preconditioner.

Detailed descriptions of AGMG (aggregation-based algebraic multigrid) preconditioners can be found in [89, 77]; we use the implementation from [88] (release 4.2.2). For DDM, we use the setting described in Example 4.1; a two-level overlapping domain decomposition preconditioner with an overlapping ratio of 0.5 is applied. Note that in the latter case, we actually solve a generalized eigenvalue problem $A - \lambda M$, see Remark 4.1, with M representing the mass matrix from the finite element method.

Table 4.1: Behavior of φ for Laplacian eigenvalue problems with AGMG and DDM preconditioners.

AGMG					
h	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}
$\cos^2\varphi$	0.0331	0.0192	0.0117	0.0064	0.0033
$1-\kappa_\nu^{-1}$	0.6200	0.6260	0.6352	0.6378	0.6390
χ	0.0534	0.0307	0.0184	0.0101	0.0051
DDM with $H = 2^{-2}$					
h	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
$\cos^2\varphi$	0.1961	0.1935	0.1915	0.1905	0.1901
$1-\kappa_\nu^{-1}$	0.8221	0.8201	0.8189	0.8182	0.8178
χ	0.2386	0.2360	0.2339	0.2328	0.2324
DDM with $h = 2^{-8}$					
H	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}
$\cos^2\varphi$	0.1901	0.0720	0.0202	0.0052	0.0013
$1-\kappa_\nu^{-1}$	0.8178	0.8213	0.8242	0.8278	0.8320
χ	0.2324	0.0877	0.0246	0.0063	0.0016

4.6.1.1 Behavior of φ

The purpose of the first experiment is to study the angle of distortion φ. A small value of $\cos\varphi$ is favorable for our theory, because this implies that the condition on the initial vector becomes loose. We let $A = -\Delta_h$ denote the discretization of the Laplacian and B denote the preconditioner. For either of the two scenarios described above, the preconditioner is only available implicitly, through matrix-vector products with B^{-1}. The ratio κ_ν can be obtained by computing the smallest and largest eigenvalues of $-B^{-1}\Delta_h$ with the Lanczos method. The definition of the angle φ requires the computation of both Bu^* and $B^{-1}u^*$. While the second computation is straightforward, the first computation is not, because B is not explicitly available. Instead of the matrix-vector multiplication Bu^*, we solve the linear system $B^{-1}z = u^*$ using the preconditioned conjugate gradient method with $-\Delta_h$ as the preconditioner.

Defining

$$\chi := \frac{\cos^2\varphi}{1-\kappa_\nu^{-1}},$$

the bound (4.3.2.2) is equivalent to $\chi \leq 1$. From the numerical results in Table 4.1, one observes that χ is significantly smaller than 1, demonstrating that the bound (4.3.2.2) is not sharp. Table 4.1 confirms our theoretical result $\cos\varphi = \mathcal{O}(H)$ from (4.5.1.4). For the AGMG preconditioner, it can be observed that $\cos^2\varphi = \mathcal{O}(h)$, i.e. a very favorable behavior.

Table 4.2: Empirical success probabilities for Laplacian eigenvalue problems with AGMG and DDM preconditioners.

AGMG					
h	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}
$\lambda(u_0) < \lambda_2$	0%	0%	0%	0%	0%
$\mathrm{dist}_B(u_0, u^*) < \varphi$	44.8%	53.7%	62.1%	67.9%	77.3%
DDM with $H = 2^{-2}$					
h	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
$\lambda(u_0) < \lambda_2$	0.5%	0%	0%	0%	0%
$\mathrm{dist}_B(u_0, u^*) < \varphi$	16.9%	7.2%	4.8%	1.3%	1.2%
DDM with $h = 2^{-8}$					
H	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}
$\lambda(u_0) < \lambda_2$	0%	0%	0%	0%	0%
$\mathrm{dist}_B(u_0, u^*) < \varphi$	0.9%	11.1%	41.1%	71.3%	85.8%

4.6.1.2 Empirical probability tests

In most practical situations, PINVIT is used with a random initial vector u_0. Therefore it is of interest to measure the empirical success probability for our condition $\mathrm{dist}_B(u_0, u^*) < \varphi$ and for the condition $\lambda(u_0) < \lambda_2$ required by [61].

It is tempting to choose a Gaussian random vector u_0, but such a choice is unfortunate—it yields an empirical success probability close to zero for both conditions. A Gaussian random vector tends to be highly oscillatory, whereas the eigenvector u^* is typically very smooth. We address this issue by using a smoother multivariate normal random vector. As the inverse Laplacian affects smoothing, it makes sense to choose $u_0 \sim \mathcal{N}(0, B^{-2})$, which can be computed as $u_0 = B^{-1}\omega$ for a Gaussian random vector ω. Using 1000 independent random trials, we report the empirical success probabilities in Table 4.2, which impressively show the superiority of our condition on the initial vector.

4.6.2 Mixed-precision preconditioners for kernel matrices

Following the setting in [64, Section 5.4], we perform experiments with the mixed-precision preconditioner from Section 4.5.2 for targeting the smallest eigenvalues of a kernel matrix. Choosing independent Gaussian random vectors $x_1, \ldots, x_n \in \mathbb{R}^n$, we consider the Laplacian kernel matrix defined by

$$(A)_{ij} = \exp\Big(\frac{-\|x_i - x_j\|}{2}\Big), \quad i, j = 1, \ldots, n.$$

Similarly, choosing another set of independent Gaussian random vectors $y_1, \ldots, y_n \in \mathbb{R}^n$ and $K(x, y) = (x^T y + 1)^3$, we consider the complex kernel matrix defined

by

$$(A)_{ij} = K(x_i, x_j) + K(y_i, y_j) + \Im\big(K(x_i, y_j) - K(y_i, x_j)\big).$$

In both cases, we choose B to be the preconditioner obtained from performing the Cholesky factorization of A in single precision. As in the previous section, we measured the empirical success probability for $\mathrm{dist}_B(u_0, u^*) < \varphi$ and $\lambda(u_0) < \lambda_2$. We choose u_0 to be a Gaussian random vector, set $n \in \{512, 1024, 2048, 4096\}$. For each n, we verify the initial conditions on u_0 by sampling 1000 independent random initial vectors, and collect the results in Table 4.3. With such effective mixed-precision preconditioners, our condition on the initial vector achieves nearly 100% success probability, whereas the condition $\lambda(u_0) < \lambda_2$ appears to be never satisfied.

Table 4.3: Empirical success probabilities for dense kernel matrices with mixed-precision preconditioner.

	Laplacian Kernel				Complex Kernel			
n	512	1024	2048	4096	512	1024	2048	4096
$\lambda(u_0) < \lambda_2$	0%	0%	0%	0%	0%	0%	0%	0%
$\mathrm{dist}_B(u_0, u^*) < \varphi$	100%	100%	100%	100%	96.8%	97.9%	100%	100%

5 A state-of-the-art eigenvalue solver and its convergence guarantees

In this and the next section, we would like to delve deeper into general eigenvalue solvers. We showcase how our theory presented in Section 2 is useful not only for analyzing the simple gradient descent version presented there, but also more advanced versions. This section follows our work [12].

5.1 Introduction

A simple idea that comes from [12] and can improve the practical performance of vanilla gradient descent is the following:

- Run iteration (2.3.0.1) choosing the step size η_t via an exact line search.
- Substitute the vanilla gradient approach with a *conjugate gradients* approach.

Our work [12] shows that if one uses a gradient update, the exact line search step is very easy and cheap to compute. Moreover, this algorithm can be shown to enjoy a local linear convergence rate using the results of Section 2. When one moves to the conjugate gradients approach, both the line search strategy lacks some theoretical rigor and convergence analysis is not possible. However, the algorithm performs extremely well in practice, which allows us to confidently say that it is state-of-the-art from a practical performance viewpoint.

As this section concerns general eigenvalue solvers, we turn again to the block case, i.e. we consider the optimization problem

$$f(\mathcal{X}) = -\tfrac{1}{2}\operatorname{Tr}(X^T A X), \text{ with } \mathcal{X} = \operatorname{Span}(X),\ X^T X = I \tag{5.1.0.1}$$

The $1/2$ scaling is harmless and is included in order to match the language of [12].

5.2 Gradient method on Grassmann

In a gradient approach we would like to produce an iterate $\mathcal{X}_{t+1} = \operatorname{Span}(X_{t+1})$ starting from $\mathcal{X}_t = \operatorname{Span}(X_t)$ following a rule of the form

$$X_{t+1} = X_t - \eta \operatorname{grad} f(\mathcal{X}_t), \tag{5.2.0.1}$$

where the step size $\eta > 0$ is this time to be determined by some line search. The direction opposite to the gradient is a direction of decrease for the objective function f. However, it is unclear what value of the step η yields the largest decrease in the value of f. This means that some care has to be exercised in the search for the optimal η.

For a Riemannian method defined on a manifold, the search direction (here, $-\operatorname{grad} f(\mathcal{X}_t)$) always lies in the tangent space of the current point (here, $\mathcal{X}_t$)

of said manifold. This makes sense since directions orthogonal to the tangent space leave the objective function constant up to first order in the step if the iterates are restricted to lie on the manifold.

As discussed in Section 2, the Riemannian gradient of the block Rayleigh quotient at $\mathcal{X} = \mathrm{Span}(X)$ is

$$\mathrm{grad}\, f(\mathcal{X}) = -P_X AX \equiv -(AX - XC), \tag{5.2.0.2}$$

with the orthogonal projector $P_X = I - XX^T$, and the projected matrix $C = X^T AX$ (notice here the lack of a factor 2 due to the $1/2$ scaling in our cost function).

Even though $-\mathrm{grad}\, f(\mathcal{X}_t)$ is in the tangent space (and a direction of decrease for f), we are not interested in X_{t+1} per se but in the subspace that it spans. In particular, since we use orthonormal bases to define the value of f on the manifold, we will need to "correct" the non-orthogonality of the update (5.2.0.1) when considering f. This will be discussed shortly. For now we establish a few simple relations.

For simplicity we denote $X := X_t$ an orthonormal basis of the current iterate $\mathcal{X}$, $\tilde{X} := X_{t+1}$ a (probably non-orthonormal) basis of the new iterate $\tilde{\mathcal{X}}$ and $G := \mathrm{grad}\, f(\mathcal{X})$ the gradient direction. Then a step of the gradient method satisfies $\tilde{X} = X - \eta G$ and we have

$$f(\tilde{\mathcal{X}}) = f(\mathcal{X}) - \eta\, \mathrm{Tr}((AX)^T P_X (AX)) - \frac{\eta^2}{2}\, \mathrm{Tr}((AX)^T P_X A P_X (AX)). \tag{5.2.0.3}$$

We also have the following relations

$$\begin{aligned} (AX)^T P_X (AX) &= -(AX)^T G = -(G^T (AX))^T = -G^T (AX) && (5.2.0.4)\\ &= (AX)^T P_X^T P_X (AX) = G^T G && (5.2.0.5) \end{aligned}$$

where the second equality exploits the fact that P_X is an orthogonal projector.

Thus, the coefficient of η in the right-hand side of (5.2.0.3) is nothing but $\|G\|_F^2$ and, therefore, as expected, the direction of G is a descent direction: for small enough η, $\tilde{X}$ will be close to orthonormal, and regardless of the value of the trace in the last term, we would get a decrease of the objective function f. This will be the case unless we have already reached a critical point where $G = 0$.

When looking at (5.2.0.3) it may appear at first that when A is SPD, it is possible to increase the value of η arbitrarily and decrease the objective function arbitrarily. This is clearly incorrect because we have not yet adjusted the basis: we need to find the subspace spanned by $\tilde{X}$ and compute the related value of the objective function. In the following we address this issue by actually optimizing the objective function in the Grassmann manifold.

Observe that since $X^T G = 0$ we have:

$$\tilde{X}^T \tilde{X} = (X - \eta G)^T (X - \eta G) = I + \eta^2 G^T G.$$

Let the spectral decomposition of G^TG be

$$G^TG = VD_\beta V^T \tag{5.2.0.6}$$

and denote $\beta = \texttt{Diag}(D_\beta)$ the eigenvalues. We now define the diagonal matrix

$$D_\eta \equiv (I + \eta^2 D_\beta)^{1/2}. \tag{5.2.0.7}$$

In order to make $\tilde{X}$ orthogonal without changing its linear span, we multiply it to the right by $VD_\eta^{-1}V^T$. This way, we obtain the matrix

$$X(\eta) = \tilde{X}VD_\eta^{-1}V^T = (X - \eta G)VD_\eta^{-1}V^T. \tag{5.2.0.8}$$

that depends on the step η and is easily seen to be orthonormal,

$$X(\eta)^T X(\eta) = D_\eta^{-1}V^T(I + \eta^2 G^TG)VD_\eta^{-1} = I.$$

While it is tempting to remove the V^T in (5.2.0.8) as this does not change the linear span, it is useful to keep it. The normalization is only then equivalent to the polar factor of $X - \eta G$. In the context of optimization on manifolds, this so-called retraction has many nice properties. In particular, $X(\eta)$ is a best approximation of $X - \eta G$ in the set of orthonormal matrices. In addition, this retraction has an easy vector transport that is invariant to the choice of representative in the subspace, which will be important later in Section 5.5, where we discuss the acceleration of the gradient method.

Remark 5.1 *A retraction in general is a first order approximation of the geodesics of a manifold (see Section 3.6 in [21]). Similarly, vector transport is some method that transports tangent vectors to a new tangent space, which consists a first order approximation of the parallel transport (Section 10.5 in [21]). The main results of this section remain the same if the retraction discussed above is substituted by the exact geodesics and the vector transport by the exact parallel transport. For simplicity though, we will keep the approximate choices. The empirical performance is more or less the same, while some retractions and vector transports are easier to compute compared to exact geodesics and the parallel transport.*

5.3 Efficient line search

We can now tackle the issue of determining the optimal η. If we set

$$X_v = XV, \qquad G_v = GV, \tag{5.3.0.1}$$

then from (5.2.0.4)–(5.2.0.5) we get the relation $G_v^T A X_v = -G_v^T G_v$. In addition, note that $G_v^T G_v = V^T G^T G V = D_\beta$. With these relations we can now show:

$$\begin{aligned} & f(\mathcal{X}(\eta)) \\ = & -\tfrac{1}{2}\,\mathrm{Tr}(V D_\eta^{-1} V^T (X-\eta G)^T A (X-\eta G) V D_\eta^{-1} V^T) \\ = & -\tfrac{1}{2}\,\mathrm{Tr}(D_\eta^{-1} (X_v-\eta G_v)^T A (X_v-\eta G_v) D_\eta^{-1}) \\ = & -\tfrac{1}{2}\,\mathrm{Tr}\left(D_\eta^{-2}\left(X_v^T A X_v + 2\eta (G_v^T G_v) + \eta^2 (G_v^T A G_v)\right)\right) \\ = & -\tfrac{1}{2}\,\mathrm{Tr}\left(\left(I+\eta^2 D_\beta\right)^{-1}\left(X_v^T A X_v + 2\eta\ D_\beta + \eta^2\ G_v^T A G_v\right)\right) \end{aligned} \tag{5.3.0.2}$$

We will simplify notation by introducing the diagonal matrices:

$$D_\alpha = \mathtt{Diag}(\alpha_1,\ldots,\alpha_p) \quad \text{with} \quad \alpha_i = (X_v^T A X_v)_{ii}, \tag{5.3.0.3}$$

$$D_\gamma = \mathtt{Diag}(\gamma_1,\ldots,\gamma_p) \quad \text{with} \quad \gamma_i = (G_v^T A G_v)_{ii}. \tag{5.3.0.4}$$

If we call u_i the left singular vector of G associated with $\sqrt{\beta_i}$ then we get the useful relation

$$\gamma_i \equiv v_i^T G^T A G v_i = \beta_i u_i^T A u_i. \tag{5.3.0.5}$$

Observe that when D is a diagonal matrix and C is arbitrary, then $\mathtt{Diag}(DC) = D\ \mathtt{Diag}(C)$. Therefore, (5.3.0.2) simplifies to:

$$f(\mathcal{X}(\eta)) = -\tfrac{1}{2}\,\mathrm{Tr}\left(\left(I+\eta^2 D_\beta\right)^{-1}\left(D_\alpha + 2\eta D_\beta + \eta^2 D_\gamma\right)\right)\ . \tag{5.3.0.6}$$

This is a rational function that is the sum of k terms corresponding to the k diagonal entries of the matrix involved in (5.3.0.6):

$$f(\mathcal{X}(\eta)) = -\frac{1}{2}\sum_{i=1}^{k} \frac{\alpha_i\ + 2\beta_i\eta + \gamma_i\eta^2}{1+\beta_i\eta^2}. \tag{5.3.0.7}$$

When $\eta \to \infty$ each term $\frac{\alpha_i\ +2\beta_i\eta+\gamma_i\eta^2}{1+\beta_i\eta^2}$ will *decrease* to its limit γ_i/β_i. The derivative of $f(\mathcal{X}(\eta))$ satisfies

$$\frac{df(\mathcal{X}(\eta))}{d\eta} = -\sum_{i=1}^{k} \frac{\beta_i\ + (\gamma_i\ - \alpha_i\ \beta_i)\eta - \beta_i^2\eta^2}{(1+\beta_i\eta^2)^2}\ . \tag{5.3.0.8}$$

This derivative is the negative sum of k branches each associated with a diagonal entry of the matrix of which the trace is taken in the above equation. The numerator $\beta_i + (\gamma_i - \alpha_i\beta_i)\eta - \beta_i^2\eta^2$ of each branch has the shape of an inverted parabola and has a negative and a positive root. Therefore, the derivative (5.3.0.8) is nonpositive at zero[11] and as η increases away from the origin, each

[11] It is equal to $-\sum \beta_i = -\|G\|_F^2$

of the branches will have a negative derivative. The derivative remains negative until η reaches the second root which is

$$\xi_i = \frac{(\gamma_i \ - \alpha_i\beta_i) + \sqrt{(\gamma_i \ - \alpha_i\beta_i)^2 + 4\beta_i^3}}{2\beta_i^2} \ > \ 0. \tag{5.3.0.9}$$

Let $\xi_{min} = \min_i\{\xi_i\}$ and $\xi_{max} = \max_i\{\xi_i\}$. Clearly all branches of (5.3.0.7), and therefore also their sum, will decrease in value when η goes from zero to ξ_{min}. Thus, the value of the objective function (5.3.0.7) will decrease. Similarly, when η increases from ξ_{max} to infinity, the objective function (5.3.0.7) will increase. The minimal value of (5.3.0.7) with respect to η can therefore be determined by seeking the minimum in the interval $[\xi_{min}, \ \xi_{max}]$. Since both f and its derivative are available, this can be done efficiently by any standard root finding algorithm.

The algorithm to get the optimal value for η is described in Algorithm 5.2. To obtain accurate solutions, some care is required in the numerical implementation due to floating point arithmetic. We explain this in more detail in Section 5.6.1.

Algorithm 5.1 `Riemannian Gradient Descent`(A, X)

Start: Select initial $\mathcal{X}_0 = \text{Span}(X_0)$, such that $X_0^T X_0 = I$.
for $t = 0, 1, \ldots$ **do**
  Compute $G := \text{grad}\, f(\mathcal{X}_t) = -(AX_t - X_t C_k)$ with $C_k = X_k^T A X_t$.
  if $\|G\| <$ tol **then**
    return
  end if
  Diagonalize $G^T G = V D_\beta V^T$.
  Compute D_α, D_γ from (5.3.0.3) with $X = X_t$.
  Compute η as the (approximate) minimizer (5.3.0.7) using `Get_Mu`.
  Compute X_{t+1} as the polar factor of $X_t - \eta G$ like in (5.2.0.8) and set $\mathcal{X}_{t+1} = \text{Span}(X_{t+1})$
end for

Algorithm 5.2 $\eta_{out} =$ `Get_Mu`$(D_\alpha, D_\beta, D_\gamma)$

Input: Diagonal matrices $D_\alpha, D_\beta, D_\gamma$ of (5.3.0.3).
Compute smallest root ξ_{min} and largest root ξ_{max} among the roots ξ_i of (5.3.0.9)
Compute an approximation η_{out} of the minimum of f on $[\xi_{min}, \ \xi_{max}]$ by safe-guarded root finding on (5.3.0.7).
Return: value η_{out}

5.4 Convergence of the gradient method

We start our convergence analysis by proving that the gradient method from Algorithm 5.1 converges globally to a critical point, that is, where the Riemannian gradient is zero. This result is valid for any initial iterate $\mathcal{X}_0$ but it does

not give a linear rate of convergence. Such result holds also for the algorithm presented in Section 2, but we omitted it there since the relevant paper does not contain it.

When $\mathcal{X}_0$ is close to the dominant subspace, we also prove a linear rate of convergence of the objective function. The closeness condition depends on the spectral gap δ of the dominant subspace but only as $\mathcal{O}\left(\sqrt{\delta}\right)$. This result seems to be new.

5.4.1 Global convergence of the gradient vector field

We examine the expression (5.3.0.7) in order to obtain a useful lower bound. We first rewrite (5.3.0.7) as follows:

$$\begin{aligned} f(\mathcal{X}(\eta)) &= -\frac{1}{2}\sum_{i=1}^{k}\frac{\alpha_i(1+\beta_i\eta^2)-\alpha_i\beta_i\eta^2+2\beta_i\eta+\gamma_i\eta^2}{1+\beta_i\eta^2} \\ &= -\frac{1}{2}\sum_{i=1}^{k}\alpha_i - \frac{1}{2}\sum_{i=1}^{k}\frac{2\beta_i\eta+(\gamma_i-\alpha_i\beta_i)\eta^2}{1+\beta_i\eta^2}\ . \end{aligned} \tag{5.4.1.1}$$

The first sum on the right-hand side is just the objective function before the update, that is, the value of f at the current iterate $\mathcal{X}(0) = \mathcal{X}$. The second sum depends on the step η and thus represents what may be termed the "loss" of the objective function for a given η.

Lemma 5.1 *Define $L \equiv \lambda_1(A) - \lambda_n(A)$. Then for any given $\eta \geq 0$ the "loss" term (2nd term in right-hand side of* (5.4.1.1)*) satisfies*

$$-\frac{1}{2}\sum_{i=1}^{k}\frac{2\beta_i\eta+(\gamma_i-\alpha_i\beta_i)\eta^2}{1+\beta_i\eta^2} \geq \frac{(2-L\eta)\eta}{2(1+\beta_{max}\eta^2)}\cdot\|G\|_F^2, \tag{5.4.1.2}$$

where $G = \operatorname{grad} f(\mathcal{X}(0))$ and $\beta_{max} = \max\beta_i$.

Proof We exploit (5.3.0.5) and set $\tau_i = u_i^T A u_i$ in order to rewrite the term $\gamma_i - \alpha_i\ \beta_i$ in the numerator as $\gamma_i - \alpha_i\ \beta_i = (\tau_i - \alpha_i)\beta_i$. From (5.3.0.1) and (5.3.0.4), we have $\alpha_i = x_i^T A x_i$ with $x_i = X v_i$. Hence, the term $\tau_i - \alpha_i \equiv u_i^T A u_i - x_i^T A x_i$ represents the difference between two Rayleigh quotients with respect to A and therefore, $\tau_i - \alpha_i \geq -L$. Thus the "loss" term satisfies

$$-\frac{1}{2}\sum_{i=1}^{k}\frac{2\beta_i\eta+(\gamma_i-\alpha_i\beta_i)\eta^2}{1+\beta_i\eta^2} \geq -\frac{1}{2}\sum_{i=1}^{k}\frac{2-L\eta}{1+\beta_i\eta^2}\beta_i\eta. \tag{5.4.1.3}$$

The denominators $1+\beta_i\eta^2$ can be bounded from above by $1+\beta_{max}\eta^2$ and this will result in:

$$-\frac{1}{2}\sum_{i=1}^{k}\frac{2\beta_i\eta+(\gamma_i-\alpha_i\beta_i)\eta^2}{1+\beta_i\eta^2}\geq-\frac{1}{2}\sum_{i=1}^{k}\frac{2-L\eta}{1+\beta_{max}\eta^2}\beta_i\eta=\frac{(2-L\eta)\eta}{2(1+\beta_{max}\eta^2)}\sum_{i=1}^{k}\beta_i. \tag{5.4.1.4}$$

The proof ends by noticing that $\sum_{i=1}^{k}\beta_i=\|G\|_F^2$ due to (5.2.0.6). ■

Lemma 5.2 *If η_{opt} is the optimal η obtained from a line search at a given $\mathcal{X}$, then*

$$f(\mathcal{X}(\eta_{opt}))\leq-\frac{1}{2}\sum_{i=1}^{k}\alpha_i-\frac{2}{5}\frac{\|G\|_F^2}{L}\ . \tag{5.4.1.5}$$

Proof The right-hand side (5.4.1.2) is nearly minimized for $\eta_s=1/L$, so we consider this special value of η. We have

$$f(\mathcal{X}(\eta_{opt}))\leq f(\mathcal{X}(\eta_s))\leq-\frac{1}{2}\sum_{i=1}^{k}\alpha_i-\frac{(2-L\eta_s)\eta_s}{2(1+\beta_{max}\eta_s^2)}\cdot\|G\|_F^2.$$

The second inequality in the above equation follows from (5.4.1.1) and the previous Lemma 5.1. Calculating the right-hand side for $\eta_s=1/L$ yields:

$$f(\mathcal{X}(\eta_{opt}))\leq-\frac{1}{2}\sum_{i=1}^{k}\alpha_i-\frac{\|G\|_F^2}{2(L+\beta_{max}/L)}.$$

Be Lemma 2.2, we have $\beta_{max}\leq\frac{L^2}{4}$ since β_{max} is the biggest eigenvalue of G^TG. Plugging this into the last inequality we get the desired result. ■

The property (5.4.1.5) in Lemma 5.2 is known as a sufficient decrease condition of the line search. We can now follow standard arguments from optimization theory to conclude that (Riemannian) gradient descent for the smooth objective function f converges in gradient norm.

Theorem 5.3 *The sequence of gradient matrices* $\operatorname{grad} f(\mathcal{X}_t)$ *generated by Riemanian gradient descent with exact line search converges (unconditionally) to zero starting from any X_0.*

Proof We will proceed by avoiding the use of indices. First, we observe that the traces of the iterates, that is, the consecutive values of $f(\mathcal{X}(\eta_{opt}))$ converge since they constitute a bounded decreasing sequence. Recall that the first term, that is, minus the half sum of the α_i's in the right-hand side of (5.4.1.5), is the value of the objective function at the previous iterate. Thus, the second term

in (5.4.1.5) is bounded from above by the difference between two consecutive traces:

$$0 \leq \frac{2}{5}\frac{\|G\|_F^2}{L} \leq -f(\mathcal{X}(\eta_{opt})) - \frac{1}{2}\sum_{i=1}^{k}\alpha_i = -f(\mathcal{X}(\eta_{opt})) + f(\mathcal{X}), \qquad (5.4.1.6)$$

and therefore it converges to zero. This implies that the sequence of gradients also converges to 0. ■

The bound of Lemma 5.2 can be used to prove some particular rate of convergence for the gradient vector field. This argument is again classical for smooth optimization. It is a slow (algebraic) rate but it holds for any initial guess.

Proposition 5.4 *The iterates $\mathcal{X}_t$ of Algorithm* 5.1 *satisfy*

$$\min_{t=0,\ldots,K-1} \|\operatorname{grad} f(\mathcal{X}_t)\|_F \leq \sqrt{\frac{5}{2}L(f(\mathcal{X}_0) - f^*)}\frac{1}{\sqrt{K}},$$

where f^* is the minimum of f.

Proof Since f^* is the minimum of f, it holds

$$f(\mathcal{X}_0) - f^* \geq f(\mathcal{X}_0) - f(\mathcal{X}_t) = \sum_{t=0}^{K-1}(f(\mathcal{X}_t) - f(\mathcal{X}_{t+1})). \qquad (5.4.1.7)$$

After some rearrangement, Lemma 5.2 provides the bound

$$-\frac{1}{2}\sum_{i=1}^{m}\alpha_i - f(\mathcal{X}(\eta_{opt})) = f(\mathcal{X}_t) - f(X_{t+1}) \geq \frac{2}{5L}\|\mathrm{grad} f(\mathcal{X}_t)\|_F^2.$$

Taking the sum of this inequality for $t = 0, ..., K-1$, we obtain the lower bound

$$\sum_{t=0}^{K-1}(f(\mathcal{X}_t) - f(\mathcal{X}_{t+1})) \geq K\frac{2}{5L}\min_{t=0,\ldots,K-1}\|\operatorname{grad} f(\mathcal{X}_t)\|_F^2.$$

Combining with (5.4.1.7) gives the desired result. ■

5.4.2 Local linear convergence

The previous proposition establishes a global but slow convergence to a critical point. We now turn to the question of proving a fast (linear) rate to the dominant k-dimensional subspace $\mathcal{V}_\alpha = \mathrm{Span}(V_\alpha)$ of A. The result will only hold locally, however, for an initial guess X_0 sufficiently close to $\mathcal{V}_\alpha$. We therefore also assume a non-zero spectral gap $\delta = \lambda_k - \lambda_{k+1} > 0$.

For showing such linear rate, we use the properties of the block Rayleigh quotient proved in Section 2. In order to guarantee a uniform lower bound for $a(\mathcal{X}_t)$ at the iterates $\mathcal{X}_t$ of Algorithm 5.1, we need to start from a distance at most $\mathcal{O}\left(\sqrt{\delta}\right)$ from the optimum.

Proposition 5.5 *An iterate $\mathcal{X}_{t+1}$ of Algorithm 5.1 starting from a point $\mathcal{X}_t$ satisfies*

$$f(\mathcal{X}_{t+1}) - f^* \leq \left(1 - \frac{8}{5} c_Q a^2(\mathcal{X}_t) \frac{\delta}{L}\right) (f(\mathcal{X}_t) - f^*).$$

Proof The result follows simply by combining the bounds of Lemma 5.2 and Proposition 2.8. By Lemma 5.2, we have

$$f(\mathcal{X}_{t+1}) - f^* \leq f(\mathcal{X}_t) - f^* - \frac{2}{5L} \| \operatorname{grad} f(\mathcal{X}_t) \|^2.$$

By the PL property of f in Proposition 2.8, we have

$$\begin{aligned} f(\mathcal{X}_{t+1}) - f^* &\leq f(\mathcal{X}_t) - f^* - \frac{8}{5} c_Q a^2(\mathcal{X}_t) \frac{\delta}{L} (f(\mathcal{X}_t) - f^*) \\ &\leq \left(1 - \frac{8}{5} c_Q a^2(\mathcal{X}_t) \frac{\delta}{L}\right) (f(\mathcal{X}_t) - f^*). \end{aligned}$$

This provides the desired result. ∎

The convergence factor in the previous theorem still involves a quantity $a(\mathcal{X}_t)$ that depends on the iterate $\mathcal{X}_t$ at step t. To get a convergence factor for all t that only depends on the initial step, we need to bound $a(\mathcal{X}_t)$ globally from below and independently of t. To that end, we need to restrict the initial guess $\mathcal{X}_0$ to a radius $\mathcal{O}\left(\sqrt{\delta}\right)$ away from the optimum. The reason for that is that, using Proposition 5.5, we can only show that function values do not increase. In order to obtain a bound for the distances of the iterates to the optimum (and thus also for $a(\mathcal{X}_t)$), we need to use the quadratic growth condition of Proposition 2.4. This leads to a loss of a factor δ in the upper bound for the squared distances of the iterates to the optimum.

Theorem 5.6 *Algorithm 5.1, where $\mathcal{X}_0$ is such that*

$$\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \sqrt{\frac{2 c_Q \delta}{L}},$$

produces iterates $\mathcal{X}_t$ that satisfy

$$f(\mathcal{X}_t) - f^* \leq \left(1 - c_Q \frac{2\delta}{5L}\right)^t (f(\mathcal{X}_0) - f^*)$$

for all $t \geq 0$.

Proof Recall that $a(\mathcal{X}_t) = \theta_t / \tan\theta_t$ with θ_t the largest principal angle between $\mathcal{X}_t$ and $\mathcal{V}_\alpha$. By the result of Proposition 5.5, we have

$$f(\mathcal{X}_{t+1}) - f^* \leq \left(1 - \frac{8}{5} c_Q a^2(\mathcal{X}_t) \frac{\delta}{L}\right) (f(\mathcal{X}_t) - f^*) \leq f(\mathcal{X}_t) - f^*,$$

since $1 - \frac{8}{5} c_Q a^2(\mathcal{X}_t) \frac{\delta}{L} \leq 1$. By induction, we can conclude that

$$f(\mathcal{X}_t) - f^* \leq f(\mathcal{X}_0) - f^*,$$

for all $t \geq 0$.

Then by quadratic growth and smoothness of f (Propositions 2.1 and 2.4), we have

$$\begin{aligned} \operatorname{dist}^2(\mathcal{X}_t, \mathcal{V}_\alpha) &\leq \frac{1}{c_Q \delta} (f(\mathcal{X}_t) - f^*) \leq \frac{1}{c_Q \delta} (f(\mathcal{X}_0) - f^*) \\ &\leq \frac{L}{2 c_Q \delta} \operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq 1, \end{aligned}$$

for all $t \geq 0$, by the assumption on the initial distance between $\mathcal{X}_0$ and $\mathcal{V}_\alpha$.

By elementary properties of $\cos(x)$ and $x/\tan(x)$ and using (1.3.1.7), we have

$$a(\mathcal{X}_t) \geq \cos(\theta_k(\mathcal{X}_t, \mathcal{V}_\alpha)) \geq \cos(\operatorname{dist}(\mathcal{X}_t, \mathcal{V}_\alpha)) \geq \cos(1) \geq \frac{1}{2}.$$

Plugging this in the result of Proposition 5.5 and by an induction argument, we get the desired result. ■

Finally, we present an iteration complexity for computing an approximation of the leading eigenspace via Algorithm 5.1. The $\tilde{\mathcal{O}}$ notation hides non-leading logarithmic factors. This result is standard when a non-asymptotic convergence rate (like the one of Theorem 5.6) is available.

Corollary 5.7 *Algorithm* 5.1 *where* X_0 *satisfies the assumption of Theorem* 5.6 *computes an estimate* $\mathcal{X}_T$ *of* $\mathcal{V}_\alpha$ *such that* $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$ *in at most*

$$T = \frac{5\pi^2 L}{8\delta} \log \frac{f(\mathcal{X}_0) - f^*}{c_Q \varepsilon \delta} + 1 = \tilde{\mathcal{O}}\left(\frac{L}{\delta} \log \frac{f(\mathcal{X}_0) - f^*}{\varepsilon}\right).$$

many iterations.

Proof For $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$, it suffices to have

$$f(\mathcal{X}_t) - f^* \leq c_Q \epsilon^2 \delta$$

by quadratic growth of f in Proposition 2.4. Using $(1-c)^t \leq \exp(-ct)$ for all $t \geq 0$ add $0 \leq c \leq 1$, Theorem 5.6 gives that it suffices to choose T as the smallest integer such that

$$f(\mathcal{X}_T) - f^* \leq \exp\left(-c_Q \frac{2\delta}{5L} T\right) (f(\mathcal{X}_0) - f^*) \leq c_Q \epsilon^2 \delta.$$

Solving for T and substituting $c_Q = 4/\pi^2$, we get the required statement. ■

5.5 Accelerated gradient method

It is natural to consider an accelerated gradient algorithm as an improvement to the standard gradient method. For convex quadratic functions on $\mathbb{R}^n$, the best example is the conjugate gradient algorithm since it speeds up convergence significantly at virtually the same cost per step as the gradient method. In our case, the objective function is defined on $\mathrm{Gr}(n, k)$ and is no longer quadratic. Hence, other ideas are needed to accelerate. While there exist a few ways to accelerate the gradient method, they all introduce some kind of momentum term and compute a new search direction P recursively based on the previous iteration.

5.5.1 Polak–Ribiere nonlinear conjugate gradients

A popular and simple example to accelerate the gradient method is by the Polak–Ribiere rule that calculates a "conjugate direction" as

$$P = G + \beta P_{\text{old}} \quad \text{with} \quad \beta = \frac{\langle G - G_{\text{old}}, G \rangle}{\langle G_{\text{old}}, G_{\text{old}} \rangle}. \tag{5.5.1.1}$$

Here, we avoid indices by calling G_{old} the old gradient (usually indexed by t) and G the new one (usually indexed by $t+1$). The inner product used above is the standard Frobenius inner product of matrices where $\langle X, Y \rangle = \mathrm{Tr}(Y^T X)$. It is typical to restart with a pure gradient step ($\beta = 0$) when P is not a descent direction and at every t_{restart} iterations for some fixed choice for t_{restart}.

When applied to objective functions defined on manifolds, two modifications are required to the Euclidean update in (5.5.1.1). First, since G_{old} is a tangent vector of $\mathcal{X}_{\text{old}}$, it needs to be "transported" to the current iterate $\mathcal{X}$ in order for the inner product $\langle G_{\text{old}}, G \rangle$ to be well defined. A simple solution is by orthogonal projection onto the tangent space [12] :

$$\beta = \frac{\left\langle G - (I - XX^T)G_{\text{old}}, G \right\rangle}{\langle G_{\text{old}}, G_{\text{old}} \rangle}.$$

Since $G = (I - XX^T)G$, we do not need to compute this projection explicitly and the formula for β in (5.5.1.1) remains valid in our case. Next, since P is required to be a tangent vector, the result in (5.5.1.1) is again projected onto the tangent space as $(I - XX^T)P$.

[12] It is known that this is a vector transport that is invariant to the choice of representative of the subspaces when the retraction on Grassmann is done via the polar factor, as we do in Alg. 5.3.

5.5.2 Line search

In order to use P instead of G, we need to modify the line search in Algorithm 5.1. We will explain the differences for a general P.

Let $X(\eta) = X_{t+1}$ and $X = X_t$ denote orthonormalized bases for the new and old iterates $\mathcal{X}_{t+1}$ and $\mathcal{X}_t$. As before, we construct an iteration

$$X(\eta) = (X - \eta P)M$$

where the search direction P is a tangent vector, $P^T X = 0$, and gradient-related, $\operatorname{Tr}(G^T P) > 0$ with $G = \operatorname{grad} f(\mathcal{X})$. In addition, M is a normalization matrix such that $X(\eta)^T X(\eta) = I$.

A small calculation shows that the same normalization idea for M from the gradient method (when $P = G$) can be used here: from the eigenvalue decomposition

$$V D_\beta V^T = P^T P$$

we define

$$D_\eta = (I + \eta^2 D_\beta)^{1/2}.$$

Then it is easy to verify that

$$X(\eta) = (X - \eta P) V D_\eta^{-1} V^T \tag{5.5.2.1}$$

has orthonormal columns and represents again the polar factor of $X - \eta P$.

Let $P_v = PV$ and $X_v = XV$. To perform the line search for η, we evaluate f in the new point:

$$\begin{aligned} f(\mathcal{X}(\eta)) &= -\tfrac{1}{2} \operatorname{Tr}(D_\eta^{-1} V^T (X - \eta P)^T A (X - \eta P) V D_\eta^{-1}) \\ &= -\tfrac{1}{2} \operatorname{Tr}(D_\eta^{-1} (X_v - \eta P_v)^T A (X_v - \eta P_v) D_\eta^{-1}) \\ &= -\tfrac{1}{2} \operatorname{Tr}\left(D_\eta^{-2} \left(X_v^T A X_v - 2\eta (P_v^T A X_v) + \eta^2 (P_v^T A P_v)\right)\right) \\ &= -\tfrac{1}{2} \operatorname{Tr}\left(\left(I + \eta^2 D_\beta\right)^{-1} \left(D_\alpha + 2\eta\ D_\zeta + \eta^2\ D_\gamma\right)\right) \end{aligned} \tag{5.5.2.2}$$

where

$$\begin{aligned} &D_\alpha = \mathtt{Diag}(X_v^T A X_v), \quad D_\beta = \mathtt{Diag}(P_v^T P_v), \\ &D_\gamma = \mathtt{Diag}(P_v^T A P_v), \quad D_\zeta = -\mathtt{Diag}(P_v^T A X_v). \end{aligned} \tag{5.5.2.3}$$

Comparing to (5.3.0.6), we see that a new D_ζ has appeared. Observe that $D_\alpha, D_\beta, D_\gamma$ all have non-negative diagonal but this is not guaranteed for D_ζ. If $P = G$, then $-P_v^T A X_v = P_v^T P_v$ and thus $D_\zeta = D_\beta$. For a gradient related P that is a tangent vector, we know that $0 \leq \operatorname{Tr}(P^T G) = -\operatorname{Tr}(V P^T P_X A X V) = -\operatorname{Tr}(P_v^T A X_v) = \operatorname{Tr}(D_\zeta)$. However, that does not mean that all the diagonal entries of D_ζ are non-negative, only their sum is. This lack of positive diagonal complicates the line search, as we will discuss next.

Let $\alpha_i, \beta_i, \gamma_i, \zeta_i$ be the ith diagonal entry of $D_\alpha, D_\beta, D_\gamma, D_\zeta$, resp. The rational function that represents (5.5.2.2) and generalizes (5.3.0.7) satisfies

$$f(\mathcal{X}(\eta)) = -\frac{1}{2}\sum_{i=1}^{k} \frac{\alpha_i \ + 2\zeta_i\eta + \gamma_i\eta^2}{1+\beta_i\eta^2}, \tag{5.5.2.4}$$

with derivative

$$\frac{df(\mathcal{X}(\eta))}{d\eta} = -\sum_{i=1}^{k} \frac{\zeta_i \ + (\gamma_i \ - \alpha_i \ \beta_i)\eta - \beta_i\zeta_i \ \eta^2}{(1+\beta_i\eta^2)^2} \ . \tag{5.5.2.5}$$

Since we do not know the sign of ζ_i, each term in (5.5.2.5) has a quadratic in the numerator that can be convex or concave. This is different from (5.3.0.8), where it is always convex (accounting for the negative sign outside the sum) since $\zeta_i = \beta_i$. In this case, there is a term with a concave quadratic and we can therefore not directly repeat the same arguments for the bracketing interval of η based on the zeros of the quadratics in (5.5.2.5). When there are negative ζ_i's, we could restart the iteration and replace P by the gradient G. Since this wastes computational work, we prefer to simply disregard the branches that are concave when determining the bracket interval.

Overall, the line search for the CG approach will cost a little more than that for the gradient method, since we have an additional (diagonal) matrix to compute, namely D_ζ.

Algorithm 5.3 `Riemannian Conjugate Gradient Descent`(A, X)

Start: Select initial $\mathcal{X}_0 = \text{Span}(X_0)$ such that $X_0^T X_0 = I$. Set $G = P = 0$.
for $t = 0, 1, \ldots$ **do**
  Keep $G_{\text{old}} := G$.
  Update $G := \text{grad}\, f(\mathcal{X}_t) = -(AX_t - X_t C_t)$ with $C_t = X_t^T A X_t$.
  if $\|G\| <$ tol **then**
    return
  end if
  Diagonalize $G^T G = V D_\beta V^T$.
  Compute $D_\alpha, D_\beta, D_\gamma, D_\zeta$ from (5.5.2.3) with $X = X_t$.
  Compute $\beta = \langle G - G_{\text{old}}, G\rangle \,/\, \langle G_{\text{old}}, G_{\text{old}}\rangle$
  Update $P := (I - X_t X_t^T)(G + \beta P)$
  if restart **then**
    $P := G$
  end if
  Compute η as the minimizer of (5.5.2.4) using a modified version `Get_Mu`.
  Compute X_{t+1} as the polar factor of $X_t - \eta P$ like in (5.5.2.1), and set $\mathcal{X}_{t+1} = \text{Span}(X_{t+1})$.
end for

5.6 Numerical implementation and experiments

5.6.1 Efficient and accurate implementation

A proper numerical implementation of Algorithms 5.1 and 5.3, and in particular the line search, is critical to obtain highly accurate solutions. We highlight here four important aspects.

In addition, we give some details on how to improve the efficiency of a direct implementation of these algorithms so that they require the same number of matrix vector products with A, as subspace iteration and LOBCG.

Calculation of bracket The β_i's in (5.3.0.9) can be very small in some situations. If we set $\delta_i = \gamma_i - \alpha_i\beta_i$ then cancellation may cause loss of accuracy in formula (5.3.0.9) when $\delta_i < 0$. We can circumvent this by observing that in this case:

$$\xi_i = \frac{\sqrt{\delta_i^2 + 4\beta_i^3} - |\delta_i|}{2\beta_i^2} = \frac{4\beta_i^3}{2\beta_i^2(|\delta_i| + \sqrt{\delta_i^2 + 4\beta_i^3})} = \frac{2}{|\delta_i/\beta_i| + \sqrt{(\delta_i/\beta_i)^2 + 4\beta_i}}. \tag{5.6.1.1}$$

When $\delta_i > 0$ we can simply use (5.3.0.9) which we rewrite as

$$\xi_i = \frac{1}{2\beta_i}\left(\frac{\delta_i}{\beta_i} + \sqrt{\left(\frac{\delta_i}{\beta_i}\right)^2 + 4\beta_i}\right). \tag{5.6.1.2}$$

Calculation of the minimizer For numerical reasons, it is advisable to compute a root of grad f instead of a minimum of f. This can be done in an effective way by a safe-guarded root finding algorithm, like the Dekker–Brent algorithm from `fzero` in MATLAB. Since this algorithm converges superlinearly, we rarely need more than 10 function evaluations to calculate the minimizer of f in double precision.

Efficient matvecs At each iteration t, the line search requires AP_t and AX_t; see (5.5.2.3). Supposing that AX_t was calculated previously, it would seem that we need another multiplication of A with P_t which is not needed in subspace iteration (accelerated by Chebyshev or not). Fortunately, it is possible to avoid one of these multiplications. First, we proceed as usual by computing the next subspace $\mathcal{X}_{t+1}$ from the polar decomposition

$$X_{\text{new}} = (X - \eta P)VD_\eta^{-1}V^T.$$

Instead of calculating AX_{new} explicitly in the next iteration, we observe that

$$AX_{\text{new}} = (AX - \eta AP)VD_\eta^{-1}V^T. \tag{5.6.1.3}$$

Hence, it suffices to compute only AP explicitly at each iteration since AX can be updated by the recursion above.

Except for a small loss of accuracy when the method has nearly converged, this computation behaves very well numerically. In practice, the product AX is only calculated explicitly when $\eta = O(\varepsilon_{\text{mach}})$.

Efficient orthonormalization The line search procedure requires the diagonalization $P^TP = VD_\beta V^T$, which has a non-negligible cost of $O(nk^2 + k^3)$ flops. Fortunately, the result of this decomposition can be used again for the normalization of X_{new} by the polar factor, as explained in (5.2.0.8) and (5.5.2.1). Compared to using QR for the normalization, there is therefore very little overhead involved.

5.6.2 Comparison with subspace iteration for a Laplacian matrix

We first test our methods for the standard 2D finite difference Laplacian on a 35×40 grid, resulting in a symmetric positive definite matrix of size $n = 1400$. Recall that the dimension of the dominant subspace to be computed is denoted by k.

Algorithms 5.1 and 5.3 (with $t_{\text{restart}} = 75$ are compared to subspace iteration applied to a shifted and scaled matrix $(A-cI)/h$ and a filtered matrix $p_d(A)$ with given degree d, with $p_d(x) = C_d((x-c)/h)$ where C_d is a Chebyshev polynomial of degree d. The shift c and scaling h are discussed briefly in Section 1.2. More precisely, we consider $c = (\lambda_{k+1} + \lambda_n)/2$ and $h = (\lambda_{k+1} - \lambda_n)/2$. See also [122] for a concrete implementation based on a three-term recurrence that only requires computing one product AX_t per iteration. These choices of the shift and the polynomial are in some sense optimal for the given degree d. In addition, we compared to the locally optimal block conjugate gradients method (LOBCG) from [59] which is closely related to Riemannian CG but with a higher cost per iteration; see Section 5.6.4 for more details.

Observe that both subspace iteration methods make use of the exact values of the smallest eigenvalue λ_n and of the largest unwanted eigenvalue λ_{k+1}. While this is not a realistic scenario in practice, the resulting convergence behavior should therefore be seen as the best case possible for those methods. Algorithms 5.1 and 5.3 on the other hand, do not require any knowledge on the spectrum of A and can be applied immediately.

The subspace iteration with Chebyshev acceleration will restart every d iterations to perform a normalization of X_t and, in practice, adjusts the Chebyshev polynomial based on refined Ritz values[13]. For small d, the method does not enjoy as much acceleration as for large d. On the other hand, for large d the method is not stable.

In Figure 5.1, the convergence of the objective function $f(\mathcal{X}_t)$ is visible for subspace dimension $k = 6$ and polynomial degrees $d \in \{15, 30, 60\}$. All methods perform per iteration only one block matvec of A with a matrix of size

[13]This is not done in our numerical tests since we supply the method the exact unwanted spectrum.

$n \times k$. Since this is the dominant cost in large-scale eigenvalue computations like SCF, we plotted the convergence in function of this number[14].

The benefits of acceleration by the Chebyshev polynomial filter or by Riemannian CG are clearly visible in the figure. In black lines, we also indicated the asymptotic convergence $O(\gamma^t)$ in function of the number of matvecs t for two values of γ. In particular, it is well known (see our results in Section 2) that

$$\kappa = \frac{\lambda_1 - \lambda_n}{\lambda_k - \lambda_{k+1}} = \mathcal{O}(1/\delta). \tag{5.6.2.1}$$

is the condition number of the Riemannian Hessian of f at the dominant subspace $\mathcal{V}_\alpha$ with spectral gap δ. From this, the asymptotic convergence rate of Riemannian GD is known (see [73, Chap. 12.5]) to satisfy

$$\gamma_{GD} = \left(\frac{\kappa - 1}{\kappa + 1}\right)^2 = 1 - \mathcal{O}(\delta).$$

In addition, for Riemannian CG we conjecture the rate

$$\gamma_{CG} = \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right)^2 = 1 - \mathcal{O}\left(\sqrt{\delta}\right)$$

based on the similarity to classical CG for a quadratic objective function with condition number κ. For both Algorithms 5.1 and 5.3, we see that the actual convergence is very well predicted by the estimates above.

5.6.3 A few other matrices

As our next experiment, we apply the same algorithms from the previous section (but without restarting to have parameter free Riemannian methods) to a few different matrices and several choices for the subspace dimension k. In addition, we target also the minimal eigenvalues by applying the methods to $-A$ instead of A. This is not a problem, as the Riemannian gradient of f on Grassmann is invariant under shifts. More concretely, Riemannian gradient descent (RGD) and Riemannian CG (RCG) with exact line search applied to $-A$ produce the same iterates as when applied to $-A + cI$, for any $c \in \mathbb{R}$. For Algorithm 5.3 the signs of G and G_{old} flip, but the parameter β remains the same at each iteration. Thus, both methods converge to the eigenvectors associated to the largest eigenvalues of $-A$, which are the eigenvectors associated with the smallest eigenvalues of A.

Except for the standard finite difference matrices for the 3D Laplacian, the matrices used were taken from the SuiteSparse Matrix Collection [32]. This results in problems with moderately large Riemannian condition numbers κ, defined in (5.6.2.1).

[14]For this example with very sparse A, the SI methods are much faster per iteration than the Riemannian methods. This is mainly because SI only needs to orthonomalize every d times.

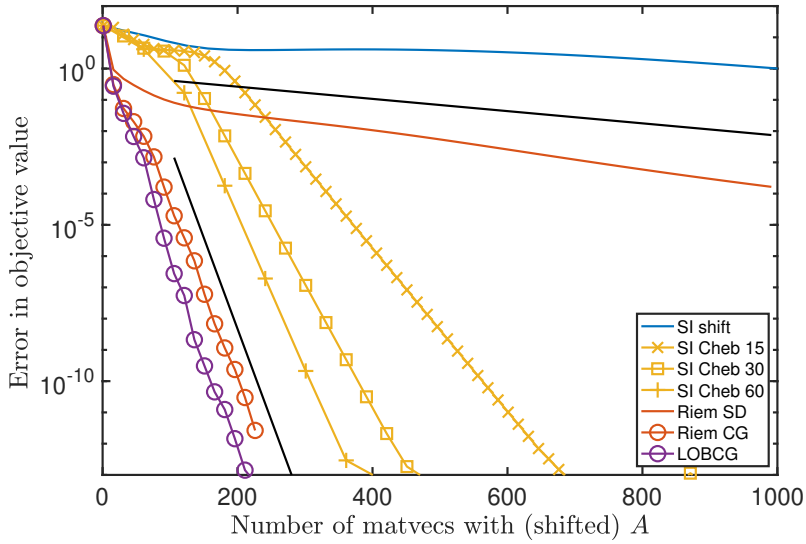


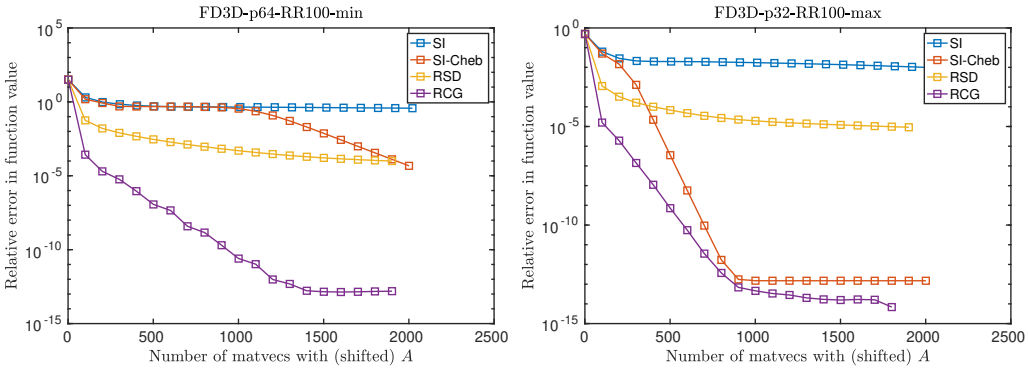
Figure 5.1: Error in objective value for subspace iteration (SI), Riemannian gradient descent (GD), Riemannian nonlinear conjugate gradients (CG), and locally optimal block conjugate gradients (LOBCG) for a Laplacian matrix of size $n = 1400$ based on finite differences when computing the dominant subspace of dimension $k = 6$. For SI, optimal shift and optimal Chebyshev polynomials were used of various degree (number in legend). The black lines estimate the asymptotic convergence speed as explained in the text.

Due to the larger size of some of these matrices, we first compute with a Krylov–Schur method (implemented in MATLAB as `eigs`) the eigenvalues that are required to determine the optimal Chebyshev filter in subspace iteration. The Riemannian methods do not require this or any other information. As optimal value f^* for the function value, we took the best value of the results computed from all methods, including the Krylov–Schur method.

FD3D This matrix is the 3D analogue of the matrix we tested in the previous section. It corresponds to a standard finite difference discretization of the Laplacian in a box with zero Dirichlet boundary conditions. We used $n_x = 35, n_y = 40, n_z = 25$ points in the x, y, z direction, resp. The resulting matrix is of size 35000. Compared to the earlier experiment, we took larger subspace dimensions and also a minimization of the Rayleigh quotient. All these elements make for a more challenging problem numerically.

problem	type	dimension k	Riem. cond. nb.	Cheb. degree
1	min	64	$3.53 \cdot 10^4$	100
2	max	32	$5.54 \cdot 10^3$	100

In Fig. 5.2, we see that the convergence of the maximization problem is very similar to that of the 2D case, although the asymptotic convergence rate of



Problem nb. 1 Problem nb. 2

Figure 5.2: The FD3D matrix.

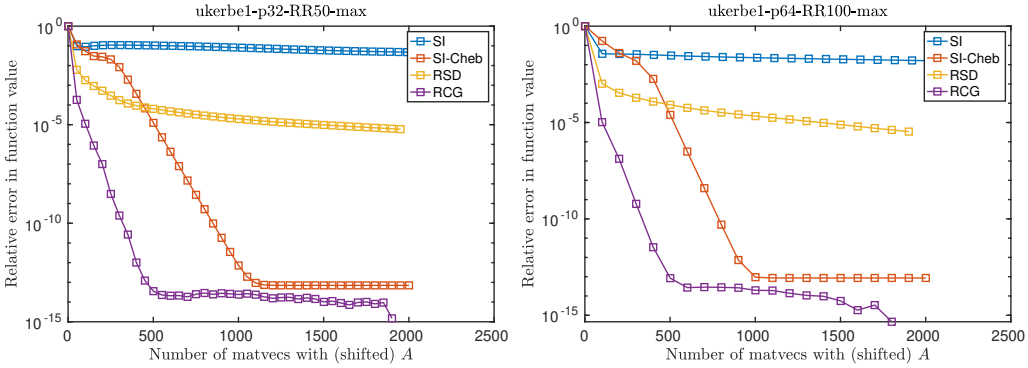
Riemannian CG seems to be slower than that of subspace iteration with optimal filter. This can be improved by restarting (not shown) but even without it, the results are good. On the other hand, the more relevant case of finding the minimal eigenvalues of a Laplacian matrix turns out to be a challenge for SI with or without Chebyshev acceleration. In fact, even with a degree 100 polynomial it takes about 1000 iterations before we see any acceleration. The Riemannian methods, on the other hand, converge much faster and already from the first iterations.

ukerbe1 This matrix is related to a 2D finite element problem on a locally refined grid and it has a relatively small size $n = 5981$. It is therefore more interesting than the uniform grid of the Laplacian examples above. We tested the following parameters.

problem	type	dimension k	Riem. cond. nb.	Cheb. degree
3	max	32	$4.85 \cdot 10^3$	50
4	max	64	$5.21 \cdot 10^3$	100

In Figure 5.3, we observe that the Riemannian algorithms converge faster than their subspace iteration counterparts. This behavior is seen for many choices of p and the Chebyshev degree. Since the spectrum of this matrix is symmetric around zero, the min problems are mathematically equivalent to the max problems, and therefore omitted.

ACTIVSg70K We now test a larger matrix of size 69999. It models a synthetic (yet realistic) power system grid from the Texas A&M Smart Grid Center. This



Problem nb. 3 Problem nb. 4

Figure 5.3: The ukerbe1 matrix.

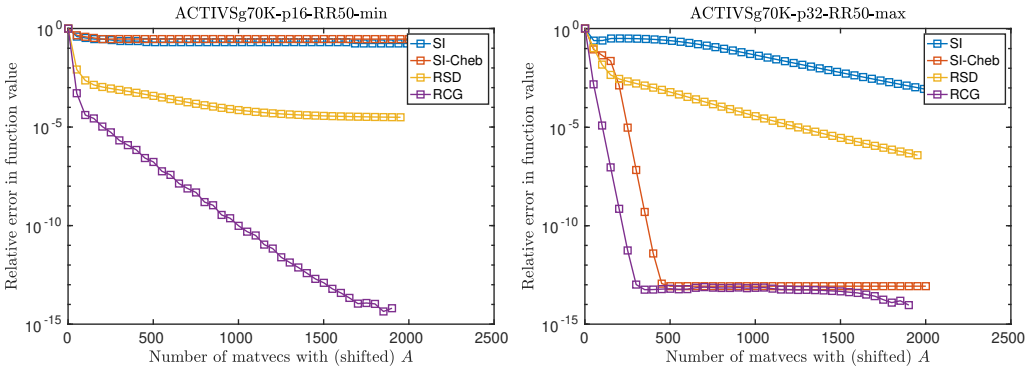
matrix has a spectral gap of $\mathcal{O}(10)$ but the Riemannian condition number, which represents the correct relative measure of difficulty, is still large. Such a different kind of scale makes this an interesting matrix to test our algorithms.

problem	type	dimension k	Riem. cond. nb.	Cheb. degree
5	min	16	$1.15 \cdot 10^4$	50
6	max	32	$1.29 \cdot 10^3$	50

For the minimization problem (nb. 5), we see that the Riemannian algorithms converge considerably faster than subspace iteration with or without Chebyshev acceleration of degree 50. (The reason for the bad performance of the Chebyshev acceleration is due to numerical instability with a degree 50 polynomial for this problem.) For the maximization problem (nb. 6), Riemannian CG and Chebyshev acceleration with degree 50 have very similar asymptotic convergence speed although the Riemannian algorithm has a faster start. The same conclusion hods for Riemannian GD and standard subspace iteration, although their convergence is of course significantly slower.

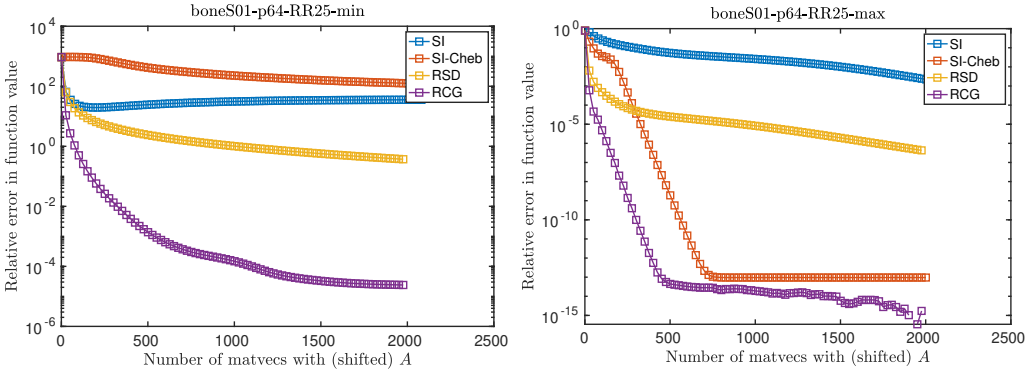
boneS01 This final matrix is part of the Oberwolfach model order reduction benchmark set and models a 3D trabecular bone. It is our largest example of size $n = 127224$. As we can see from the table below, for subspace dimension $k = 64$ the minimization problem is particularly challenging with a large Riemannian condition number.

problem	type	dimension k	Riem. cond. nb.	Cheb. degree
7	min	64	$2.57 \cdot 10^6$	25
8	max	64	$2.05 \cdot 10^3$	25



Problem nb. 5 Problem nb. 6

Figure 5.4: The ACTIVSg70K matrix.



Problem nb. 7 Problem nb. 8

Figure 5.5: The boneS01 matrix.

The convergence of the methods is visible in Fig. 5.5. We can make similar observations as for the example above: the Riemannian algorithms have a faster initial convergence compared to the subspace variants. In addition, the accelerated variants are clear improvements.

5.6.4 Comparison to LOBCG

It is instructive to compare the Riemannian CG method from Alg. 5.3 to the locally optimal block CG method (LOBCG) from [59] since both methods minimize the partial trace function f using momentum terms. LOBCG is equivalent to the better known LOBPCG method where the preconditioner is not used (i.e. set to be identity).

Let t be the iteration number. The essential difference between the two methods is that LOBCG minimizes f over all orthonormal matrices that lie in

the $3k^2$-dimensional subspace[15]

$$\mathcal{V}_t = \operatorname{Span}(X_t, G_t, X_{t-1}) = \{X_t\Omega + G_t\Psi + X_{t-1}f \colon \Omega, \Psi, f \in \mathbb{R}^{p\times p}\}. \quad (5.6.4.1)$$

Here, the residual $G_t = AX_t - X_tX_t^TAX_t$ is also the Riemannian gradient of f at $\mathcal{X}_t$. Contrary to most optimization problems, this subspace search can be computed exactly for the symmetric eigenvalue problem by the Rayleigh–Ritz procedure: the optimal solution is related to the top k eigenvectors of the symmetric $3k \times 3k$ matrix $Q_t^TAQ_t$ with Q_t an orthonormal basis for $\mathcal{V}_t$.

In contrast, the Riemannian CG method minimizes f for the scalar α during the line search applied to the orthonormalization of $X_t - \alpha P_t$. When $k > 1$, there is no explicit solution for the optimal α in terms of a smaller eigenvalue problem, but as explained above, it can be solved efficiently by diagonalizing the matrix $X_t^TX_t$.

When started at the same X_t and X_{t-1}, LOBCG will produce a basis X_{t+1} for a subspace $\mathcal{X}_{t+1}$ with a smaller objective value $f(\mathcal{X}_{t+1})$ than the Riemannian CG method. This is because an iterate produced with the step $X_t - \alpha P_t$ from Riemannian CG is contained in the subspace searched by LOBCG. It is therefore reasonable to expect[16] that LOBCG converges faster overall in terms of number of iterations.

We prove here that Riemannian CG with $t \geq 1$ is suboptimal compared to LOBCG when started at the same X_t and X_{t-1}. The case $t = 1$ is also explained in [3, Sections 4.6.5 and 8.3]. This improvement is of course more computationally expensive.

Since Riemannian CG produces iterates of the form

$$X_{t+1} = (X_t - \alpha_t P_t)M_t \quad (5.6.4.2)$$

with M_t the normalization so that X_{t+1} has orthonormal columns, it is clear that

$$X_{t+1} \in \operatorname{Span}(X_t, P_t).$$

Here, $\operatorname{Span}(\cdot,\cdot)$ is to be interpreted as in (5.6.4.1), i.e. as a subspace of dimension $2k$. Since $P_t = (I - X_tX_t^T)(G_t + \beta_t P_{t-1})$, we also have $\operatorname{Span}(P_t) \subseteq \operatorname{Span}(G_t, P_{t-1}, X_t)$, from which it follows that

$$X_{t+1} \in \operatorname{Span}(X_t, G_t, P_{t-1}).$$

The relation (5.6.4.2) also shows that $P_{t-1} \in \operatorname{Span}(X_{t-1}, X_t)$ if M_{t-1} is invertible, which is true generically. We therefore get that

$$X_{t+1} \in \operatorname{Span}(X_t, G_t, X_{t-1}) = \mathcal{V}_t,$$

[15]When X_t converges, adding X_{t-1} to the columns of X_t and G_t would lead to numerical cancellation when computing an orthonormal basis for $\mathcal{V}_t$. In the implementation of LOBCG, a different matrix is therefore added that has better numerical properties. For theoretical investigations, we can ignore it.

[16]Since the iteration is not stationary and depends on the previous iterates, one cannot conclude that LOBCG always produces iterates with lower objective value than Riemannian CG.

where $\mathcal{V}_t$ is the subspace used in LOBCG. Since LOBCG is optimal for f over all orthonormal matrices with k columns in $\mathcal{V}_t$, it will be a lower bound of $f(\mathcal{X}_{t+1})$.

In Table 5.1, we have compared LOBCG to Riemannian CG (denoted by RCG) for the same matrices we tested above. For the matrices ukerbe1 and FD3D, we see that LOBCG indeed requires less iterations than Riemannian CG, usually by about a factor two. However, this does not mean that LOBCG is faster in terms of computational time due to an increased cost per iteration. In addition, the differences between LOBCG and Riemannian CG are less predictable for the other matrices. Overall, Riemannian CG is usually faster in computational time and also more reliable.

The increased cost per iteration of LOBCG compared to Riemannian CG is due to the additional computations for the subspace search. While both methods only require one product of the form AZ with an $n \times k$ matrix Z, LOBCG performs 3 orthonormalizations (by Cholesky decomposition) whereas Riemannian CG needs 2 (by polar factor). Furthermore, LOBCG needs 14 matrix products of the form $Y^T Z$ for $n \times k$ matrices Y and Z, while Riemannian CG requires only 4. Finally, the calculation of X_{t+1} (and AX_{t+1}) based on the coefficients from the Rayleigh–Ritz procedure is not negligible in LOBCG with a cost comparable to a product $Y^T Z$. For Riemannian CG, it is simply a linear combination of two matrices (before normalization). In our experiments, one iteration of LOBPCG was therefore about 2 to 3 times more expensive, depending on A and k.

We have also tested a version of LOBCG where all the block entries in $Q_t^T A Q_t$ are explicitly calculated (denoted by LOBCG(+) in the table). The original code replaces $X_t^T A X_t$ by the eigenvalues obtained in the Rayleigh–Ritz procedure. While this behaves well early on, we have noticed stability issues in our experiments. Figure 5.6 is a clear example where the original version of LOBCG either does not converge or behaves erratically. In other examples (not shown), the residual even grows in an unbounded way. The version LOBCG(+) is however not always an improvement over LOBCG, which can be seen from the table. This shows that an accurate implementation of CG-based methods is not trivial, even with subspace search.

For Riemannian CG, we also tested a version (denoted by RCG(+)) where the product of AX_t is explicitly calculated instead of being computed recursively as in (5.6.1.3). The unchanged number of iterations in Table 5.1 shows that there is no loss of accuracy when utilizing the recursion. When the matrix A is very sparse, like in FD3D, the version RCG(+) is less costly per iteration but for other matrices, the original version of RCG is preferable.

Table 5.1: Comparison of LOBCG and Riemannian CG (denoted by RCG) when minimizing/maximizing the partial trace for a few test matrices with different block sizes p. The time in seconds (rounded to nearest integer) and number of iterations to reach a relative residual $\|G_t\|_\infty/\|G_0\|_\infty$ of 10^{-8} is indicated in sec. and its., resp. If the method did not reach the required tolerance in 10000 iterations, a star * is given. The methods indicated with a (+) are variants that aim to be more accurate; see the text for their definition.

		LOBCG		LOBCG(+)		RCG(+)		RCG	
	problem	secs.	its.	secs.	its.	secs.	its.	secs.	its.
ACTIVSg70K	k=16 max	*	*	5	152	4	301	**3**	301
	k=16 min	**4**	152	**4**	152	29	2151	25	2151
	k=32 max	110	2152	10	152	**9**	451	11	451
	k=32 min	5	102	*	*	**4**	201	5	201
	k=64 max	*	*	*	*	**70**	1301	86	1301
	k=64 min	401	2852	**19**	102	27	551	36	551
FD3D	k=16 max	**9**	752	10	752	11	1801	10	1801
	k=16 min	7	602	8	652	3	451	**2**	451
	k=32 max	16	552	16	552	**11**	1051	**11**	1051
	k=32 min	22	802	**20**	702	37	3701	40	3701
	k=64 max	51	752	57	802	**22**	901	27	901
	k=64 min	55	802	52	752	**36**	1401	42	1401
boneS01	k=16 max	**25**	352	27	352	39	501	**25**	501
	k=16 min	276	4202	287	4002	340	4401	**209**	4401
	k=32 max	42	252	52	302	22	301	**19**	301
	k=32 min	648	4202	825	5202	480	6601	**412**	6601
	k=64 max	*	*	170	402	**101**	651	**101**	651
	k=64 min	*	*	*	*	*	*	*	*
ukerbe1	k=16 max	**1**	452	**1**	502	**1**	601	**1**	601
	k=16 min	**1**	452	**1**	452	**1**	501	**1**	501
	k=32 max	2	552	2	502	**1**	651	2	651
	k=32 min	2	402	2	402	2	701	**1**	701
	k=64 max	4	352	4	352	**3**	651	4	651
	k=64 min	5	502	5	452	**3**	551	**3**	551

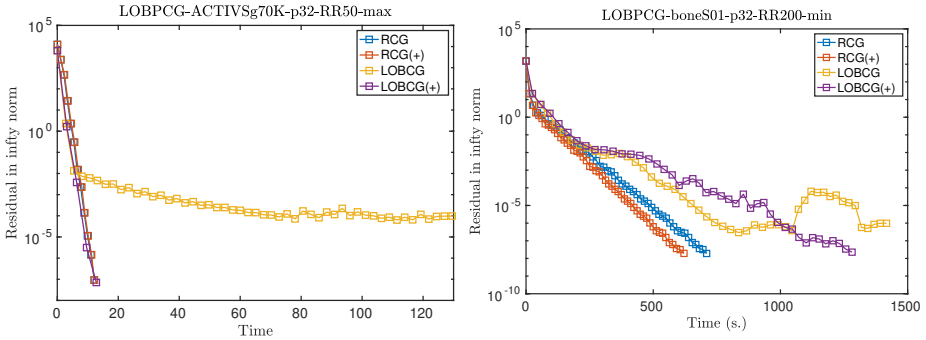


Figure 5.6: Instability of the original LOBCG method.

6 Nesterov's accelerated gradient descent for the symmetric eigenvalue problem

In this section, we examine the theoretical analysis of a Riemannian gradient descent algorithm with Nesterov momentum, in order to tackle the symmetric eigenvalue problem. We follow in general the exposition of our work [16].

6.1 Introduction

Our contribution here is the theoretical and experimental analysis of a version of Nesterov's accelerated gradient descent [79] on the Grassmann manifold for calculating a subspace spanned by the k leading eigenvectors of a matrix $A \in \mathbb{R}^{n \times n}$. To that end, we rely on the rich literature of general Riemannian algorithms, and more specifically on the formulation of Riemannian accelerated gradient descent by [119]. The other part of our analysis relies on the geodesic convexity characterization of the block Rayleigh quotient f on the Grassmann manifold, analyzed in Section 2 (Theorem 2.7). Despite that the estimate sequences technique of [119] targets only geodesically *strongly convex* objectives, there is already a technique to design estimate sequences for a weakly-quasi-strongly convex function in the Euclidean regime due to [25]. Thus, from a technical standpoint, we need to merge the Riemannian approach for strongly convex and the Euclidean approach for weakly-quasi-strongly convex functions. To that end, the geodesic search technique for selecting the momentum coefficient analyzed in [11] (which extended the similar Euclidean technique of [81]) will be of great help. This approach yields provable accelerated convergence guarantees for our algorithm. On the experimental side, we show that our algorithm is competitive compared to other state-of-the-art eigenvalue solvers.

Related work. Motivated by the classic work of Nesterov [79], a plethora of works focusing on accelerated methods on Riemannian manifolds has been developed in recent years. We refer the reader to [4, 119] for algorithms targeting geodesically strongly convex objective functions. There is a second line of work targeting objectives that are geodesically convex but not strongly convex with moderate success so far [11, 58]. Recent work [75] seems to give a better answer on the acceleration problem in the geodesically convex but not strongly convex setting, its complexity though makes it inaccessible to us. In all cases, the main obstacles consist of designing an estimate sequence that can handle the non-linearity of the manifold. There is also a recent line of work on no-go results on acceleration on manifolds, and namely that one cannot hope for any *global* accelerated method on a manifold of negative sectional curvatures [30, 41]. The latter are not directly applicable in our case, since we work with the Grassmann manifold, which is of nonnegative sectional curvatures.

However, they highlight the difficulties of designing accelerated methods on manifolds, and they give indications of why this could be achieved only locally.

Turning to the symmetric eigenvalue problem, the simplest method for computing eigenvectors and eigenvalues of a symmetric matrix is the subspace iteration. However, as discussed in the introduction (Section 1.1), this method is quite slow, both theoretically ($\tilde{\mathcal{O}}(1/\delta)$ iteration complexity) and practically. A significant part of research in numerical linear algebra has to do with "accelerating" vanilla methods like subspace iteration using more complicated mechanisms. The most well-known accelerated scheme for computing leading eigenvectors is the Lanczos method, which is a member of the family of Krylov methods. The Lanczos method has iteration complexity of $\tilde{\mathcal{O}}(1/\sqrt{\delta})$ and improves over subspace iteration. This method is, however, not stationary since it enlarges an approximation subspace in every step (like any Krylov method). The cost per iteration therefore grows both in time and in memory. This iteration is therefore restarted in practice. While the restarting strategy is empirically effective, it makes the method more complicated to use and analyze. In this paper, we therefore focus on methods that are accelerated versions of stationary methods, like gradient descent. They have the benefit of a constant cost per iteration.

An example of accelerating subspace iteration has been done employing the technology of Polyak's momentum (heavy ball) method by [115]. The resulting deterministic scheme of this paper is a subspace iteration with an extra momentum term that has guaranteed convergence in at most $\tilde{\mathcal{O}}(1/\sqrt{\delta})$ many iterations, if the momentum coefficient is chosen precisely in terms of λ_{k+1} (the $(k+1)$th largest eigenvalue). If λ_k and λ_{k+1} are not known in advance (which is usually the case), then the convergence behaviour of this algorithm can worsen considerably.

The algorithm of [115] is essentially a modern reformulation of the classical Chebyshev iteration (see [96]). An interesting contribution in [115] (except from the main contribution, a stochastic version of the algorithm) is a clever way to implement their algorithm (essentially Chebyshev iteration) in a numerically stable manner (Lemma 12), paying the extra cost of a QR-decomposition in a $(2n) \times k$ matrix (instead of $n \times k$). A different approach based on non-linear conjugate gradients is presented in Section 5. The conjugate gradient method combined with a choice of step size via an exact line search has excellent empirical performance, but it is very hard to prove any theoretical convergence guarantees (Section 5 provides theoretical guarantees only for Algorithm 5.1 and not for Algorithm 5.3). Another interesting method that is empirically accelerated but comes without much theory is LOBPCG [59].

In this section, we deviate from the previous research directions and develop a version of Nesterov's accelerated gradient descent on the Grassmann manifold for the symmetric eigenvalue problem. When measured in terms of matrix-

vector products, every iterate of this algorithm has double the cost as subspace iteration and subspace iteration with momentum [115]. The algorithm does in addition incur overheads when computing the momentum terms and the geodesic. These costs are however not dependent on A and involve only dense linear algebra routines that are typically very optimized in practical implementations.

The analysis of our method reveals that one needs at most $\tilde{\mathcal{O}}(1/\sqrt{\delta})$ many iterations to compute the dominant subspace with accuracy ϵ, if the initialization is $\mathcal{O}(\delta^{3/4})$ close to the optimal subspace. The need for local initialization is an artifact of the general analysis of the Riemannian version of accelerated gradient descent we use, developed in [119]. Also, our algorithm relies on an almost exact knowledge for the gap $\delta = \lambda_k - \lambda_{k+1}$, similarly to [115] which requires exact knowledge of λ_k and λ_{k+1}.

We do not claim that the work of this section is a go-to for practitioners. Even from a theoretical point of view, the analysis is so complicated that even the expert reader might find it difficult to follow. The convergence guarantee is not impressive either, as one needs a very good initial guess in order to achieve accelerated convergence. We do believe though that bringing the famous symmetric eigenvalue problem together with the equally famous Nesterov accelerated gradient descent algorithm merits some discussion. From a higher lever viewpoint, it is certainly interesting that this algorithm can be used for tackling this problem and perhaps this work serves as a good basis for future improvements.

6.2 Weak estimate sequence

For reasons related to both the weak nature of geodesic convexity and the non-linearity of the working domain (Grassmann manifold), we introduce a *weaker notion* of the classical estimate sequence than [79]. This is the strategy in both [25] and [119].

Definition 6.1 *A weak estimate sequence for f is a sequence of functions $(\phi_t)_{t=0}^{\infty}$ defined on the Grassmann manifold, and a sequence of positive scalars $(\tau_t)_{t=0}^{\infty}$, such that*

$$\lim_{t\to\infty} \tau_t = 0 \quad \textit{and} \quad \phi_t(\mathcal{V}_\alpha) \leq (1-\tau_t) f(\mathcal{V}_\alpha) + \tau_t \phi_0(\mathcal{V}_\alpha)$$

where $\mathcal{V}_\alpha = \operatorname{argmin}_{\mathcal{X}\in \mathrm{Gr}(n,k)} f(\mathcal{X})$. We denote such a weak estimate sequence by the pair (τ_t, ϕ_t).

The difference with the classical definition is that the inequality holds only at the optimum $\mathcal{V}_\alpha$ and not at any point.

We utilize weak estimate sequences in the following way:

Proposition 6.2 *If for some sequence of subspaces $(\mathcal{X}_t)_{t=0}^{\infty}$, we have*

$$f(\mathcal{X}_t) \leq \phi_t^* := \min_{\mathcal{X} \in \mathrm{Gr}(n,k)} \phi_t(\mathcal{X})$$

where (ϕ_t, τ_t) is a weak estimate sequence, then

$$f(\mathcal{X}_t) - f^* \leq \tau_t(\phi_0(\mathcal{V}_\alpha) - f^*).$$

Proof The proof is direct by the fact that

$$\begin{aligned} f(\mathcal{X}_t) &\leq \min_{\mathcal{X} \in \mathrm{Gr}(n,k)} \phi_t(\mathcal{X}) \leq \phi_t(\mathcal{V}_\alpha) \\ &\leq (1-\tau_t) f(\mathcal{V}_\alpha) + \tau_t \phi_0(\mathcal{V}_\alpha) = (1-\tau_t) f^* + \tau_t \phi_0(\mathcal{V}_\alpha). \end{aligned}$$

Rearranging we get the result. ■

Now, we describe how to construct a weak estimate sequence for our geodesically WQSC function f in (1.2.1.5). The result below is valid for any function that satisfies Theorem 2.7, but we phrase it directly for f for simplicity. For ease of notation we denote

$$\mu := 2c_Q\delta$$

for the rest of Section 6.

Proposition 6.3 *Let f be the block Rayleigh quotient* 1.2.1.5. *Choose an arbitrary function $\phi_0 \colon \mathrm{Gr}(n,k) \to \mathbb{R}$ and an arbitrary sequence $(\mathcal{Y}_t)_{t=0}^{\infty}$ of subspaces in $\mathrm{Gr}(n,k)$. We also choose a sequence $(\alpha_t)_{t=0}^{\infty}$ of scalars such that $\alpha_t \in (0,1)$ and $\Sigma_{t=0}^{\infty} \alpha_t = \infty$.*

Define $\tau_0 = 1$. For all $t \geq 0$, let B_t be a lower bound for $a(\mathcal{Y}_t)$ (as defined in Theorem 2.7*) and define*

$$\begin{aligned} \tau_{t+1} &:= (1-\alpha_t)\tau_t \\ \bar{\phi}_{t+1}(\mathcal{X}) &:= (1-\alpha_t)\phi_t(\mathcal{X}) \\ &\quad + \alpha_t \left(f(\mathcal{Y}_t) + \frac{1}{B_t} \langle \mathrm{grad} f(\mathcal{Y}_t), \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{X}) \rangle + \frac{\mu}{2} \mathrm{dist}^2(\mathcal{Y}_t, \mathcal{X}) \right) \end{aligned}$$

If $\phi_t(\mathcal{V}_\alpha) \leq \bar{\phi}_t(\mathcal{V}_\alpha)$ for all $t \geq 0$, then the pair (τ_t, ϕ_t) is a weak estimate sequence for f.

Proof We prove the main inequality involving ϕ_t in the definition of a weak estimate sequence by induction. For $k = 0$, we have $\phi_0(\mathcal{V}_\alpha) = (1-\tau_0)f^* + \tau_0\phi_0(\mathcal{V}_\alpha)$ because $\tau_0 = 1$. Assume that the inequality holds for some $k \geq 0$:

$$\phi_t(\mathcal{V}_\alpha) - f^* \leq \tau_t(\phi_0(\mathcal{V}_\alpha) - f^*).$$

Then, we have

$$\begin{aligned}\phi_{t+1}(\mathcal{V}_\alpha) - f^* &\leq \bar{\phi}_{t+1}(\mathcal{V}_\alpha) - f^* \\ &\leq (1-\alpha_t)\phi_t(\mathcal{V}_\alpha) + \alpha_t f^* - f^* \\ &= (1-\alpha_t)(\phi_t(\mathcal{V}_\alpha) - f^*) \\ &\leq (1-\alpha_t)\tau_t(\phi_0(\mathcal{V}_\alpha) - f^*) \\ &= \tau_{t+1}(\phi_0(\mathcal{V}_\alpha) - f^*).\end{aligned}$$

Thus, the inequality holds also for $t+1$ which concludes the induction. The first inequality follows from the construction of ϕ_t, the second by Theorem 2.7 (and $a(\mathcal{Y}_t) \geq B_t$) and the third by the induction hypothesis.

Furthermore, we observe that the assumption $\Sigma_{t=0}^{\infty}\alpha_t = \infty$ guarantees that $\lim_{t\to\infty}\tau_t = 0$, which finishes the proof. ■

6.3 Towards an algorithm

We now use Proposition 6.3 to construct a more specific weak estimate sequence:

Proposition 6.4 *Consider ϕ_t, α_t, B_t and $\mathcal{Y}_t$ as in Proposition 6.3 and let ϕ_t^* be defined as in Proposition 6.2. Choose $\phi_0(\mathcal{X}) = \phi_0^* + \frac{\gamma_0}{2}\|\mathrm{Log}_{\mathcal{Y}_0}(\mathcal{X})\|^2$ (this is possible since Proposition 6.3 is for an arbitrary ϕ_0). For $k \geq 0$, we define the following terms recursively:*

- $\bar{\gamma}_{t+1} := (1-\alpha_t)\gamma_t + \alpha_t\mu$
- $\mathcal{V}_{t+1} := \mathrm{Exp}_{\mathcal{Y}_t}\left(\frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{\alpha_t}{B_t\bar{\gamma}_{t+1}}\mathrm{grad}f(\mathcal{Y}_t)\right)$
- $\phi_{t+1}^* := (1-\alpha_t)\phi_t^* + \alpha_t f(\mathcal{Y}_t) - \frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}}\|\mathrm{grad}f(\mathcal{Y}_t)\|^2$
  $+ \frac{\alpha_t(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\left(\frac{\mu}{2}\mathrm{dist}^2(\mathcal{Y}_t,\mathcal{V}_t) + \frac{1}{B_t}\langle\mathrm{grad}f(\mathcal{Y}_t), \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\rangle\right)$

If γ_{t+1} is chosen such that

$$\gamma_{t+1}\|\mathrm{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \mathrm{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1})\|^2 \leq \bar{\gamma}_{t+1}\|\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_\alpha) - \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\|^2,$$

then the pair of sequences (τ_t, ϕ_t) defined by

$$\begin{aligned}&\phi_t(\mathcal{X}) := \phi_t^* + \frac{\gamma_t}{2}\|\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2 \\ &\tau_0 = 1, \;\; \tau_{t+1} := (1-\alpha_t)\tau_t\end{aligned}$$

is a weak estimate sequence for f.

Proof We firstly prove that if $\phi_t(\mathcal{X}) = \phi_t^* + \frac{\gamma_t}{2}\|\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2$, then $\bar{\phi}_{t+1}(\mathcal{X}) = \phi_{t+1}^* + \frac{\bar{\gamma}_{t+1}}{2}\|\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\|^2$, where $\bar{\phi}_{t+1}$ is defined recursively from ϕ_t as in Proposition 6.3.

Indeed, we have

$$\bar{\phi}_{t+1}(\mathcal{X}) = (1-\alpha_t)\phi_t(\mathcal{X}) + \alpha_t\left(f(\mathcal{Y}_t) + \frac{1}{B_t}\langle \operatorname{grad} f(\mathcal{Y}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\rangle + \frac{\mu}{2}\operatorname{dist}^2(\mathcal{Y}_t, \mathcal{X})\right),$$

by its definition in Proposition 6.3. We can rewrite the right hand side as

$$\begin{aligned}
&(1-\alpha_t)\phi_t(\mathcal{X}) + \alpha_t\left(f(\mathcal{Y}_t) + \frac{1}{B_t}\langle \operatorname{grad} f(\mathcal{Y}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\rangle + \frac{\mu}{2}\operatorname{dist}^2(\mathcal{Y}_t, \mathcal{X})\right) = \\
&(1-\alpha_t)\left(\phi_t^* + \frac{\gamma_t}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2\right) \\
&+ \alpha_t\left(f(\mathcal{Y}_t) + \frac{1}{B_t}\langle \operatorname{grad} f(\mathcal{Y}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\rangle + \frac{\mu}{2}\operatorname{dist}^2(\mathcal{Y}_t, \mathcal{X})\right),
\end{aligned}$$

where we use the induction hypothesis for ϕ_t. By rearranging the terms and completing the square, we can write

$$\begin{aligned}
&(1-\alpha_t)\left(\phi_t^* + \frac{\gamma_t}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2\right) \\
&+ \alpha_t\left(f(\mathcal{Y}_t) + \frac{1}{B_t}\langle \operatorname{grad} f(\mathcal{Y}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\rangle + \frac{\mu}{2}\operatorname{dist}^2(\mathcal{Y}_t, \mathcal{X})\right) \\
&= \frac{\bar{\gamma}_{t+1}}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\|^2 + \langle \frac{\alpha_t}{B_t}\operatorname{grad} f(\mathcal{Y}_t) - (1-\alpha_t)\gamma_t \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X})\rangle \\
&+ (1-\alpha_t)\left(\phi_t^* + \frac{\gamma_t}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2\right) + \alpha_t f(\mathcal{Y}_t) \\
=&\frac{\bar{\gamma}_{t+1}}{2}\left\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \left(\frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{\alpha_t}{B_t\bar{\gamma}_{t+1}}\operatorname{grad} f(\mathcal{Y}_t)\right)\right\|^2 \\
&- \frac{\bar{\gamma}_{t+1}}{2}\left\|\frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{\alpha_t}{B_t\bar{\gamma}_{t+1}}\operatorname{grad} f(\mathcal{Y}_t)\right\|^2 \\
&+ (1-\alpha_t)\left(\phi_t^* + \frac{\gamma_t}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2\right) + \alpha_t f(\mathcal{Y}_t).
\end{aligned}$$

Plugging in the definition of $\mathcal{V}_{t+1}$ and splitting the norm in the second summand, we can write the last expression as

$$\begin{aligned}
&\frac{\bar{\gamma}_{t+1}}{2}\left\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\right\|^2 - \frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}}\|\operatorname{grad} f(\mathcal{Y}_t)\|^2 \\
&+ \frac{\alpha_t(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\left(\frac{\mu}{2}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2 + \frac{1}{B_t}\langle \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t), \operatorname{grad} f(\mathcal{Y}_t)\rangle\right) \\
&+ (1-\alpha_t)\phi_t^* + \alpha_t f(\mathcal{Y}_t).
\end{aligned}$$

Finally, we can use the definition of ϕ_{t+1}^* from ϕ_t^* and rewrite again in the desired form:

$$\phi_{t+1}^* + \frac{\bar{\gamma}_{t+1}}{2}\left\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\right\|^2.$$

This proves our first claim.

Using the previous computation and the definition of γ_{t+1}, we immediately get

$$\phi_{t+1}(\mathcal{V}_\alpha) \leq \bar{\phi}_{t+1}(\mathcal{V}_\alpha).$$

Thus, we have shown that the sequence ϕ_t defined as

$$\phi_t(\mathcal{X}) := \phi_t^* + \frac{\gamma_t}{2}\|\mathrm{Log}_{\mathcal{Y}_t}(\mathcal{X}) - \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\|^2$$

satisfies all the assumptions of Proposition 6.3. We therefore conclude that (τ_t, ϕ_t) is a weak estimate sequence. ■

Now we have a concrete definition of $\mathcal{V}_{t+1}$ from $\mathcal{Y}_t$ and $\mathcal{V}_t$. It remains to define $\mathcal{Y}_t$ through $\mathcal{X}_t$ and $\mathcal{V}_t$ and $\mathcal{X}_{t+1}$ through $\mathcal{Y}_t$. We do this with criterion to guarantee $f(\mathcal{X}_t) \leq \phi_t^*$. In order to guarantee that, let us assume that $f(\mathcal{X}_t) \leq \phi_t^*$ and see what happens with ϕ_{t+1}^* (towards having an induction step). Using the definition of ϕ_{t+1}^* in Proposition 6.4 and $f(\mathcal{X}_t) \leq \phi_t^*$, we have:

$$\begin{aligned}\phi_{t+1}^* \geq (1-\alpha_t)f(\mathcal{X}_t) + \alpha_t f(\mathcal{Y}_t) - \frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}}\|\mathrm{grad}f(\mathcal{Y}_t)\|^2 \\ + \frac{\alpha_t(1-\alpha_t)\gamma_t}{B_t\bar{\gamma}_{t+1}}\langle \mathrm{grad}f(\mathcal{Y}_t), \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\rangle.\end{aligned}$$

The usual way to proceed from here is to assume that f is geodesically convex and linearize it from below (see [119], Lemma 6). Since the Rayleigh quotient on Grassmann is only-locally convex, we employ a different strategy using a geodesic search as in [11], inspired by [81]. Namely, if we find a way to choose $\mathcal{Y}_t$ from $\mathcal{X}_t$ and $\mathcal{V}_t$ such that

$$f(\mathcal{X}_t) + \frac{\alpha_t\gamma_t}{B_t\bar{\gamma}_{t+1}}\langle \mathrm{grad}f(\mathcal{Y}_t), \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\rangle \geq f(\mathcal{Y}_t) \tag{6.3.0.1}$$

then the previous inequality can be reduced to

$$\phi_{t+1}^* \geq f(\mathcal{Y}_t) - \frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}}\|\mathrm{grad}f(\mathcal{Y}_t)\|^2. \tag{6.3.0.2}$$

Inequality (6.3.0.1) is satisfied if we choose $\mathcal{Y}_t$ through an exact search in the geodesic connecting $\mathcal{V}_t$ and $\mathcal{X}_t$:

Lemma 6.5 *Let*

$$\mathcal{Y}_t := \mathrm{Exp}_{\mathcal{V}_t}(\beta_t \mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t))$$

where

$$\beta_t := \mathrm{argmin}_{\beta\in[0,1]}(\mathrm{Exp}_{\mathcal{V}_t}(\beta \mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t))).$$

Then we have

$$f(\mathcal{Y}_t) \leq f(\mathcal{X}_t) \quad \textit{and} \quad \langle \mathrm{grad}f(\mathcal{Y}_t), \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\rangle \geq 0.$$

Proof See Appendix B in [11]. ■

Thus, if we choose $\mathcal{Y}_t$ in the manner of Lemma 6.5, the initial inequality for ϕ^*_{t+1} implies inequality (6.3.0.2). Inequality (6.3.0.2) is similar to the function value reduction that is obtained via a gradient step:

If we choose

$$\mathcal{X}_{t+1} = \mathrm{Exp}_{\mathcal{Y}_t}\left(-\frac{1}{L}\mathrm{grad} f(\mathcal{Y}_t)\right), \tag{6.3.0.3}$$

then, by L-smoothness, we have

$$f(\mathcal{X}_{t+1}) \leq f(\mathcal{X}_t) - \frac{1}{2L}\|\mathrm{grad} f(\mathcal{X}_t)\|^2$$

and if we also choose α_t such that

$$\frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}} = \frac{1}{2L},$$

then

$$f(\mathcal{X}_{t+1}) \leq f(\mathcal{Y}_t) - \frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}}\|\mathrm{grad} f(\mathcal{Y}_t)\|^2 \tag{6.3.0.4}$$

and consequently

$$f(\mathcal{X}_{t+1}) \leq \phi^*_{t+1}.$$

We have also the freedom to make the gradient step from $\mathcal{Y}_t$ following the QR-retraction (Retr) used in Section 5 and the step-size via an exact line-search:

$$\mathcal{X}_{t+1} = \mathrm{Retr}_{\mathcal{Y}_t}\left(-\eta_{\text{optimal}} \cdot \mathrm{grad} f(\mathcal{Y}_t)\right). \tag{6.3.0.5}$$

As analyzed in Section 5 this exact line search is cheap to compute.

Lemma 5.2 implies that in this case

$$f(\mathcal{X}_{t+1}) \leq f(\mathcal{X}_t) - \frac{2}{5L}\|\mathrm{grad} f(\mathcal{X}_t)\|^2$$

and this means that if we choose α_t such that

$$\frac{\alpha_t^2}{2B_t^2\bar{\gamma}_{t+1}} = \frac{2}{5L} =: \frac{1}{2\tilde{L}},$$

we have again

$$f(\mathcal{X}_{t+1}) \leq \phi^*_{t+1}$$

is guaranteed. Here $\tilde{L}$ is defined as $\frac{5}{4}L$, where L is the smoothness constant.

Choosing $\phi^*_0 = f(\mathcal{X}_0)$, we can now prove by induction that the following algorithm produces iterates $\mathcal{X}_t$, such that $f(\mathcal{X}_t) \leq \phi^*_t$, where ϕ^*_t is defined recursively as in Proposition 6.4. This analysis yields us naturally to an

algorithm, which can be proved to have accelerated convergence guarantees. We choose to write the algorithm using equation (6.3.0.3) to perform the gradient step for obtaining $\mathcal{X}_{t+1}$ from $\mathcal{Y}_t$, but we could also use equation (6.3.0.5). The only thing that changes is the constant L to $\tilde{L}$.

Significant effort needs to be spent in proving that all operations in Algorithm 6.1 are well-defined. For that to happen, we need to insure that there is a unique geodesic connecting $\mathcal{V}_t$ and $\mathcal{X}_t$, that $\|\text{grad}f(\mathcal{Y}_t)\|_2 < \frac{\pi}{2}$ and that $\left\| \frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}} \text{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{2\alpha_t}{\bar{\gamma}_{t+1}} \text{grad}f(\mathcal{Y}_t) \right\|_2 < \frac{\pi}{2}$ (i.e. these tangent vectors are inside the injecitivity domain, recall equation (1.3.1.5)). To guarantee these bounds, a careful selection of hyperparameters is crucial. Algorithm 6.1 is written in a form, in which it is not clear whether some steps are doable, for instance it is not clear whether one can find a γ_{t+1} from $\bar{\gamma}_{t+1}$ such that the requirements of step 9 are satisfied. For the moment we assume that all these can be done and we show in the next section that a careful selection of γ_0 indeed yields all the requirements of Algorithm 6.1.

Algorithm 6.1 Accelerated Gradient Descent for the Block Rayleigh Quotient

1: Initialize at $\mathcal{X}_0 = \mathcal{V}_0 \in \text{Gr}(n,p)$ and choose γ_0, such that $\frac{\mu}{2} \leq \gamma_0 \leq L$

2: **for** $k \geq 0$ **do**

3: $\quad \beta_k = \underset{\beta \in [0,1]}{\text{argmin}} \left\{ f(\text{Exp}_{\mathcal{V}_t}(\beta \text{Log}_{\mathcal{V}_t}(\mathcal{X}_t))) \right\}$

4: $\quad \mathcal{Y}_t = \text{Exp}_{\mathcal{V}_t}(\beta_k \text{Log}_{\mathcal{V}_t}(\mathcal{X}_t))$

5: $\quad 4\alpha_t^2 = \frac{(1-\alpha_t)\gamma_t + \alpha_t \mu}{L}$

6: $\quad \mathcal{X}_{t+1} = \text{Exp}_{\mathcal{Y}_t}\left(-\frac{1}{L}\text{grad}f(\mathcal{Y}_t)\right)$

7: $\quad \bar{\gamma}_{t+1} = (1-\alpha_t)\gamma_t + \alpha_t \mu$

8: $\quad \mathcal{V}_{t+1} = \text{Exp}_{\mathcal{Y}_t}\left(\frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}} \text{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{2\alpha_t}{\bar{\gamma}_{t+1}} \text{grad}f(\mathcal{Y}_t) \right)$

9: $\quad \gamma_{t+1} \|\text{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \text{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1})\|^2 \leq \bar{\gamma}_{t+1} \|\text{Log}_{\mathcal{Y}_t}(\mathcal{V}_\alpha) - \text{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\|^2$, such that $\bar{\gamma}_{t+1} \geq \gamma_{t+1} \geq \mu/2$.

10: **end for**

Remark Notice that Algorithm 6.1 is invariant under shifts of the matrix A with multiples of the identity matrix. Indeed, the only steps that are affected by such a shift are steps 3, 6 and 8, which feature the function f or its gradient. The shift $A + \alpha I$ yields a function value which is just shifted by a constant, thus step 3 remains unchanged. Also, the Riemannian gradient remains exactly the same as the orthogonal projection neutralizes the extra term obtained by the shift. Thus, steps 6 and 8 also remain unchanged. Notice that the parameters γ and μ remain unchanged as well.

Theorem 6.6 *If $\mathcal{X}_0$ satisfies*

$$\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{1}{8}\sqrt{c_Q}\left(\frac{\delta}{L}\right)^{3/4},$$

and step 9 in Algorithm 6.1 *can be satisfied, i.e. there is γ_{t+1} satisfying the required bounds, then the following holds:*

(i) $f(\mathcal{X}_t), f(\mathcal{Y}_t) \leq f(\mathcal{X}_0)$, *for all* $k \geq 0$

(ii) $a(\mathcal{Y}_t) \geq \frac{1}{2}$

(iii) The operations in steps 3, 4, 6,and 8 in Algorithm 6.1 *are well-defined in the sense that the related tangent vectors are inside the injectivity domain of* $\operatorname{Gr}(n, k)$

(iv) $f(\mathcal{X}_t) \leq \phi_t^*$, *for all* $t \geq 0$, *where* ϕ_t^* *is defined as*

$$\begin{aligned}
&\phi_0^* = f(\mathcal{X}_0)\\
&\phi_{t+1}^* = (1-\alpha_t)\phi_t^* + \alpha_t f(\mathcal{Y}_t) - \frac{\alpha_t^2}{8\bar{\gamma}_{t+1}}\|\operatorname{grad} f(\mathcal{Y}_t)\|^2\\
&\quad + \frac{\alpha_t(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}}\left(\frac{\mu}{2}\operatorname{dist}^2(\mathcal{Y}_t, \mathcal{V}_t) + 2\langle \operatorname{grad} f(\mathcal{Y}_t), \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_t)\rangle\right).
\end{aligned}$$

Proof We proceed to the proof of all points together by induction.

For $t = 0$, the first holds trivially since $\mathcal{X}_0 = \mathcal{Y}_0$.

The second point holds, because

$$a(\mathcal{Y}_0) \geq \cos(\theta_{max}(\mathcal{Y}_0, \mathcal{V}_\alpha)) \geq \cos(\operatorname{dist}(\mathcal{Y}_0, \mathcal{V}_\alpha)) \geq \cos(1) > \frac{1}{2}.$$

Here θ_{max} is used to denote the biggest principal angle between subspaces. This inequality implies that B_0 can be chosen to be $\frac{1}{2}$.

The third holds since $\mathcal{X}_0 = \mathcal{V}_0$ (steps 3-4 are well-defined) and by L-smoothness of f, we have

$$\|\operatorname{grad} f(\mathcal{Y}_0)\| \leq L\operatorname{dist}(\mathcal{Y}_0, \mathcal{V}_\alpha) = L\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{L}{8}\left(\frac{\delta}{L}\right)^{3/4} \leq \frac{L}{8}.$$

This implies that the biggest singular value of $-\frac{1}{L}\operatorname{grad} f(\mathcal{Y}_0)$ is less than $\frac{\pi}{2}$, thus $-\frac{1}{L}\operatorname{grad} f(\mathcal{Y}_0)$ is inside the injectivity domain and step 6 is well-defined for $t = 0$. For step 8, we have that $\operatorname{Log}_{\mathcal{Y}_0}(\mathcal{V}_0) = 0$, thus we need to bound $\left\|-\frac{2\alpha_0}{\bar{\gamma}_1}\operatorname{grad} f(\mathcal{Y}_0)\right\|_2$. For that, we use inequality (6.3.0.4), which can be rewritten (with $B_0 = \frac{1}{2}$) as

$$\frac{2\alpha_0^2}{\bar{\gamma}_1}\|\operatorname{grad} f(\mathcal{Y}_0)\|^2 \leq f(\mathcal{Y}_0) - f(\mathcal{X}_1) \leq f(\mathcal{X}_0) - f^*.$$

By multiplying both sides of the previous inequality with $2/\bar{\gamma}_1$ and L-smoothness of f, we get

$$\frac{4\alpha_0^2}{\bar{\gamma}_1^2}\|\text{grad} f(\mathcal{Y}_0)\|^2 \leq \frac{L}{\bar{\gamma}_1}\text{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq 2\frac{L}{\mu}\frac{1}{64}c_Q\left(\frac{\delta}{L}\right)^{\frac{3}{2}} = \frac{1}{64}\frac{L}{\delta}\left(\frac{\delta}{L}\right)^{\frac{3}{2}} \leq 1.$$

The second inequality follows from $\bar{\gamma}_1 \geq \gamma_0 \geq \frac{\mu}{2}$.

This bound implies that $-\frac{2\alpha_0}{\bar{\gamma}_1}\text{grad} f(\mathcal{Y}_0)$ is inside the injectivity domain and step 8 is well-defined for $t = 0$.

The fourth point holds trivially since ϕ_0^* is defined as $f(\mathcal{X}_0)$.

Now, we assume that all the points hold for all iterations up to iteration t and wish to prove that they still hold for $t + 1$.

By the construction of the algorithm we have

$$f(\mathcal{Y}_{t+1}) \leq f(\mathcal{Y}_t).$$

This is because $f(\mathcal{Y}_{t+1}) \leq f(\mathcal{X}_{t+1})$ (due to the geodesic search, step 3-4) and $f(\mathcal{X}_{t+1}) \leq f(\mathcal{Y}_t)$ (due to the gradient step, step 6). Thus, we can conclude that $f(\mathcal{Y}_{t+1}) \leq f(\mathcal{Y}_t) \leq f(\mathcal{Y}_0) = f(\mathcal{X}_0)$ by the induction hypothesis. The same inequalities imply that $f(\mathcal{X}_{t+1}) \leq f(\mathcal{X}_t)$ and by the induction hypothesis we have $f(\mathcal{X}_{t+1}) \leq f(\mathcal{X}_0)$. Thus, the first point is correct at iteration $t + 1$.

We can use the result of the first point to bound the distance of the iterates $\mathcal{Y}_{t+1}, \mathcal{X}_{t+1}$ from the optimum $\mathcal{V}_\alpha$, using the quadratic growth condition (Proposition 2.4):

$$\begin{aligned}\text{dist}^2(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha) &\leq \frac{1}{c_Q\delta}(f(\mathcal{Y}_{t+1}) - f^*) \leq \frac{1}{c_Q\delta}(f(\mathcal{X}_0) - f^*) \\ &\leq \frac{L}{2c_Q\delta}\text{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{1}{128}\left(\frac{\delta}{L}\right)^{1/2}.\end{aligned}$$

The same bound holds also for $\text{dist}(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)$. Thus, we have

$$\text{dist}(\mathcal{X}_{t+1}, \mathcal{V}_\alpha), \text{dist}(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha) \leq \frac{1}{8\sqrt{2}}\left(\frac{\delta}{L}\right)^{1/4}.$$

This lower bound implies that the quantity $a(\mathcal{Y}_{t+1})$ can be bounded as

$$a(\mathcal{Y}_{t+1}) \geq \cos(\theta_{max}(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha)) \geq \cos(\text{dist}(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha)) \geq \cos(1) > \frac{1}{2},$$

which implies that the second point holds at the $t + 1$ iteration.

The result of the second point together with the induction hypothesis provide that $a(\mathcal{Y}_i) \geq \frac{1}{2}$ for all $i = 0, ..., t + 1$. This means that B_i can be taken equal to $\frac{1}{2}$ in all the analysis of Sections 4 and 5 and with a choice of α_t as in step 5 of the algorithm and ϕ_{t+1}^* as defined in the statement of the fourth point, we

have automatically that $f(\mathcal{X}_{t+1}) \leq \phi^*_{t+1}$. Thus the fourth point is correct at iteration $t+1$.

For showing that the steps 3-4 and 6 in the algorithm are well-defined in iteration $t+1$ (third point), we need also a bound for $\operatorname{dist}(\mathcal{V}_{t+1}, \mathcal{V}_\alpha)$, which turns out to be a quite complicated. For that, we start by using the second bound of Proposition 1.15:

$$\operatorname{dist}(\mathcal{V}_{t+1}, \mathcal{V}_\alpha) \leq \| \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1}) \|.$$

The quantity on the right hand side is directly related to the sequence ϕ^*_t:

$$\| \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1}) \|^2 = \frac{2}{\gamma_{t+1}}(\phi_{t+1}(\mathcal{V}_\alpha) - \phi^*_{t+1}) \leq \frac{2}{\gamma_{t+1}}(\phi_0(\mathcal{V}_\alpha) - f^*) =$$
$$\frac{2}{\gamma_{t+1}}\left(\phi^*_0 + \frac{\gamma_0}{2}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) - f^*\right) = \frac{2}{\gamma_{t+1}}\left(f(\mathcal{X}_0) - f^* + \frac{\gamma_0}{2}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)\right) \leq$$
$$\frac{2}{\gamma_{t+1}}\frac{L+\gamma_0}{2}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) = \frac{L+\gamma_0}{\gamma_{t+1}}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq 4\frac{L}{\mu}\operatorname{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \leq$$
$$= 4\frac{L}{2c_Q\delta}\frac{1}{64}c_Q\left(\frac{\delta}{L}\right)^{3/2} = \frac{1}{32}\left(\frac{\delta}{L}\right)^{1/2}.$$

The first equality is implied by the definition of ϕ_{t+1}, the first inequality by Proposition 6.2 combined with the inequality $\phi^*_{t+1} \geq f(\mathcal{X}_{t+1}) \geq f^*$ which holds since we have already explained that the fourth point holds for $t+1$, the second equality by the definition of ϕ_0, the third equality by the definition of ϕ^*_0, the second inequality by L-smoothness, the third inequality by the upper bound on γ_0 and the lower bound on γ_{t+1}, and the rest are simple substitutions involving the bound in the initial distance $\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha)$.

Thus

$$\operatorname{dist}(\mathcal{V}_{t+1}, \mathcal{V}_\alpha) \leq \frac{1}{4\sqrt{2}}\left(\frac{\delta}{L}\right)^{1/4}.$$

Combining the bounds on $\operatorname{dist}(\mathcal{X}_{t+1}, \mathcal{V}_\alpha)$ and $\operatorname{dist}(\mathcal{V}_{t+1}, \mathcal{V}_\alpha)$ with the triangle inequality, we get

$$\operatorname{dist}(\mathcal{X}_{t+1}, \mathcal{V}_{t+1}) \leq \operatorname{dist}(\mathcal{X}_{t+1}, \mathcal{V}_\alpha) + \operatorname{dist}(\mathcal{V}_{t+1}, \mathcal{V}_\alpha) \leq \frac{1}{2}.$$

This implies that there is a unique geodesic connecting $\mathcal{X}_{t+1}$ and $\mathcal{V}_{t+1}$, thus steps 3-4 are well-defined in iteration $t+1$.

In addition, L-smoothness of f implies that

$$\|\operatorname{grad} f(\mathcal{Y}_{t+1})\| \leq L\operatorname{dist}(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha) \leq \frac{L}{8}$$

which provides the bound

$$\left\| -\frac{1}{L}\operatorname{grad} f(\mathcal{Y}_{t+1}) \right\|_2 \leq \frac{1}{8}.$$

The last bound implies that $-\frac{1}{L}\mathrm{grad} f(\mathcal{Y}_{t+1})$ is inside the injectivity domain, thus step 6 is well-defined in iteration $t+1$.

We lastly deal with step 8. We have that $\frac{(1-\alpha_{t+1})\gamma_{t+1}}{\bar{\gamma}_{t+2}} = \frac{(1-\alpha_{t+1})\gamma_{t+1}}{(1-\alpha_{t+1})\gamma_{t+1}+\alpha_{t+1}\mu} \leq 1$ and $\|\operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1})\| \leq \|\operatorname{Log}_{\mathcal{X}_{t+1}}(\mathcal{V}_{t+1})\| \leq \frac{1}{2}$. For the second summand $\frac{2\alpha_{t+1}}{\bar{\gamma}_{t+2}}\mathrm{grad} f(\mathcal{Y}_{t+1})$, we use inequality (6.3.0.4), which can be rewritten (with $B_{t+1} = \frac{1}{2}$) as

$$\frac{2\alpha_{t+1}^2}{\bar{\gamma}_{t+2}}\|\mathrm{grad} f(\mathcal{Y}_{t+1})\|^2 \leq f(\mathcal{Y}_{t+1}) - f(\mathcal{X}_{t+2}) \leq f(\mathcal{X}_0) - f^*.$$

Multiplying both sides with $\frac{2}{\bar{\gamma}_{t+2}}$ and using L-smoothness of f, we get

$$\frac{4\alpha_{t+1}^2}{\bar{\gamma}_{t+2}^2}\|\mathrm{grad} f(\mathcal{Y}_{t+1})\|^2 \leq \frac{L}{\bar{\gamma}_{t+2}}\mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)$$

By definition, we have $\frac{\mu}{2} \leq \gamma_{t+1} \leq \bar{\gamma}_{t+2}$. Plugging in the assumed bound on the initial distance, we get

$$\frac{4\alpha_{t+1}^2}{\bar{\gamma}_{t+2}^2}\|\mathrm{grad} f(\mathcal{Y}_{t+1})\|^2 \leq 2\frac{L}{\mu}\frac{1}{64}c_Q\left(\frac{\delta}{L}\right)^{\frac{3}{2}} = \frac{1}{64}\frac{L}{\delta}\left(\frac{\delta}{L}\right)^{\frac{3}{2}} = \frac{1}{64}\left(\frac{\delta}{L}\right)^{\frac{1}{2}} \leq \frac{1}{4},$$

where we used that $\mu = 2c_Q\delta$.

By triangle inequality, we get

$$\left\|\frac{(1-\alpha_{t+1})\gamma_{t+1}}{\bar{\gamma}_{t+2}}\operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1}) - \frac{2\alpha_{t+1}}{\bar{\gamma}_{t+2}}\mathrm{grad} f(\mathcal{Y}_{t+1})\right\| \leq \frac{1}{2} + \frac{1}{2} = 1,$$

thus

$$\frac{(1-\alpha_{t+1})\gamma_{t+1}}{\bar{\gamma}_{t+2}}\operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1}) - \frac{2\alpha_{t+1}}{\bar{\gamma}_{t+2}}\mathrm{grad} f(\mathcal{Y}_{t+1})$$

is inside the injectivity radius of $\mathrm{Gr}(n,k)$ and step 8 is well-defined at iteration $t+1$.

With that in order, the simultaneous induction of all four points is complete. ■

6.4 Effect of curvature/choice of parameters

In Algorithm 6.1, it is not clear whether we can choose γ_{t+1} from $\bar{\gamma}_{t+1}$ (step 9) in a tractable way, such that $\gamma_{t+1} \geq \frac{\mu}{2}$.

We start by showing that there is a way to choose γ_{t+1} from $\bar{\gamma}_{t+1}$, such that

$$\gamma_{t+1}\|\operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1})\|^2 \leq \bar{\gamma}_{t+1}\|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\|^2.$$

To that end, we need the following geometric result (Theorem 10 in [119]):

Lemma 6.7 *Let x, y, z, w be four points in a geodesically uniquely convex subset of a Riemannian manifold, with sectional curvatures in the interval $[-K, K]$ and*

$$\max\{\text{dist}(z, x), \text{dist}(w, z)\} \leq \frac{1}{4\sqrt{K}},$$

then

$$\|\text{Log}_w(x) - \text{Log}_w(y)\|^2 \leq (1 + 5K \max\{\text{dist}(z, x), \text{dist}(w, x)\}^2)\|\text{Log}_z(x) - \text{Log}_z(y)\|^2.$$

Proof See Theorem 10 in [119]. ■

If x, y, z, w are subspaces on the Grassmann manifold with the standard Riemannian structure, we can take $K = 2$ [113].

Note that $\mathcal{Y}_{t+1}$ is not yet computed at step 9, but it is to be computed exactly in the next iteration of the algorithm. However, this is not a problem, since the geometric result (Lemma 6.7) holds for any four points on a manifold of bounded sectional curvatures.

Proposition 6.8 *Choose*

$$\gamma_0 \geq \frac{\sqrt{\beta^2 + (1 + \beta)\frac{\mu}{L}} - \beta}{\sqrt{\beta^2 + (1 + \beta)\frac{\mu}{L}} + \beta} \cdot \mu$$

and $\gamma_0 \leq L$, where

$$\beta = \frac{1}{5}\sqrt{\frac{\mu}{L}}.$$

If $\mathcal{X}_0$ satisfies

$$\text{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{1}{8}\sqrt{c_Q}\left(\frac{\delta}{L}\right)^{3/4},$$

then one can choose γ_{t+1} from $\bar{\gamma}_{t+1}$ satisfying all the requirements in step 9 of Algorithm 6.1, as

$$\gamma_{t+1} = \frac{1}{1 + \beta}\bar{\gamma}_{t+1}.$$

Proof We proceed by induction. For $k = 0$, γ_0 satisfies trivially the main inequality of step 9 (because there is no $\bar{\gamma}_0$). Also $\gamma_0 \geq \frac{\mu}{2}$, since (easy to see)

$$\frac{\sqrt{\beta^2 + (1 + \beta)\frac{\mu}{L}} - \beta}{\sqrt{\beta^2 + (1 + \beta)\frac{\mu}{L}} + \beta} \geq \frac{1}{2},$$

if $\beta = \frac{1}{5}\sqrt{\frac{\mu}{L}}$.

Now we assume that we can choose γ_{t+1} as in step 9 of Algorithm 6.1 in the first t iterations. Then the result of Theorem 6.6 holds and its proof guarantees that

$$\text{dist}(\mathcal{Y}_t, \mathcal{V}_\alpha), (\mathcal{Y}_{t+1}, \mathcal{V}_\alpha) \leq \frac{1}{8\sqrt{2}}\left(\frac{\delta}{L}\right)^{1/4}.$$

This implies the weaker inequality

$$\operatorname{dist}(\mathcal{Y}_t, \mathcal{V}_\alpha), \operatorname{dist}(\mathcal{Y}_{t+1}, \mathcal{V}_\alpha) \leq \frac{1}{4\sqrt{K}},$$

where K is an upper bound of the sectional curvatures of the Grassmann manifold and it is taken equal to 2. Thus, the points $\mathcal{Y}_t, \mathcal{Y}_{t+1}, \mathcal{V}_{t+1}$ and $\mathcal{V}_\alpha$ satisfy the assumptions of Lemma 6.7. This gives

$$\|\operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_{t+1}}(\mathcal{V}_{t+1})\|^2 \leq \left(1 + 10\frac{1}{128}\left(\frac{\delta}{L}\right)^{1/2}\right) \|\operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_\alpha) - \operatorname{Log}_{\mathcal{Y}_t}(\mathcal{V}_{t+1})\|^2.$$

Thus, we can choose γ_{t+1} from $\bar{\gamma}_{t+1}$ such that

$$\frac{\bar{\gamma}_{t+1}}{\gamma_{t+1}} \geq 1 + \frac{10}{128}\left(\frac{\delta}{L}\right)^{1/2}$$

or similarly, we can take $\gamma_{t+1} = \frac{1}{1+\beta}\bar{\gamma}_{t+1}$ with

$$\beta \leq \frac{10}{128}\left(\frac{\delta}{L}\right)^{1/2}.$$

It easy to see that $\frac{1}{5}\left(\frac{\mu}{L}\right)^{1/2} \leq \frac{10}{128}\left(\frac{\delta}{L}\right)^{1/2}$, thus

$$\beta = \frac{1}{5}\left(\frac{\mu}{L}\right)^{1/2}$$

is a valid choice. Such a selection of β is important for the rest. Note that β is involved directly in the selection of γ_0 and affects the sequences γ_t and $\bar{\gamma}_t$.

We now prove with the aforementioned selections of β and γ_0 that γ_{t+1} selected in step 9 always satisfies $\gamma_{t+1} \geq \frac{\mu}{2}$.

We first show that if $\gamma_s \geq \frac{\sqrt{\beta^2+(1+\beta)\frac{\mu}{L}}-\beta}{\sqrt{\beta^2+(1+\beta)\frac{\mu}{L}}+\beta} \cdot \mu$, then $\alpha_s \geq \frac{\sqrt{\beta^2+(1+\beta)\frac{\mu}{L}}-\beta}{2}$. To that end, we use the definition of α_s at step 4 of Algorithm 6.1:

$$4\alpha_s^2 = \frac{(1-\alpha_s)\gamma_s + \alpha_s\mu}{L}.$$

The positive solution of this quadratic equation is

$$\alpha_s = \frac{(\mu-\gamma_s)\frac{1}{4L} + \sqrt{(\mu-\gamma_s)^2\frac{1}{16L^2} + \frac{\gamma_s}{L}}}{2} =: g(\gamma_s).$$

We first note that α_s is always less than 1. Indeed, this happens if and only if

$$(\mu-\gamma_s)\frac{1}{4L} + \sqrt{(\mu-\gamma_s)^2\frac{1}{16L^2} + \frac{\gamma_s}{L}} \leq 2$$

or even stronger if

$$(\mu - \gamma_s)\frac{1}{2L} + \sqrt{\frac{\gamma_s}{L}} \leq 2$$

which is equivalent with

$$d^2 - 2d - \frac{\mu}{L} \geq -4,$$

where $d := \sqrt{\frac{\gamma_s}{L}}$. Since $\frac{\mu}{L} \leq 1$, it suffices to hold

$$d^2 - 2d + 3 \geq 0,$$

which always holds. Now, we bound α_s from below.

The function g is increasing and we have

$$\alpha_s \geq g\left(\frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} + \beta} \cdot \mu\right) = g\left(\frac{C-\beta}{C+\beta} \cdot \mu\right),$$

where

$$C := \sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}}.$$

We have

$$\mu - \frac{C-\beta}{C+\beta}\mu = \frac{2\beta}{C+\beta}\mu = \frac{2\beta(C-\beta)}{C^2-\beta^2} = \frac{2\beta(C-\beta)}{(1+\beta)\frac{\mu}{L}}\mu = \frac{2L\beta(C-\beta)}{(1+\beta)}$$

and

$$\frac{C-\beta}{C+\beta}\mu = \frac{(C-\beta)^2}{C^2-\beta^2}\mu = \frac{(C-\beta)^2}{(1+\beta)\frac{\mu}{L}}\mu = \frac{L(C-\beta)^2}{1+\beta}.$$

Then,

$$\begin{aligned}
g\left(\frac{C-\beta}{C+\beta} \cdot \mu\right) &= \frac{\frac{2L\beta(C-\beta)}{(1+\beta)}\frac{1}{4L} + \sqrt{\frac{4L^2\beta^2(C-\beta)^2}{(1+\beta)^2}\frac{1}{16L^2} + \frac{L(C-\beta)^2}{1+\beta}\frac{1}{L}}}{2} \\
&= \frac{\frac{\beta(C-\beta)}{2(1+\beta)} + \sqrt{\frac{\beta^2(C-\beta)^2}{4(1+\beta)^2} + \frac{(C-\beta)^2}{1+\beta}}}{2} \\
&= (C-\beta)\frac{\frac{\beta}{2(1+\beta)} + \sqrt{\frac{\beta^2+4\beta+4}{4(1+\beta)^2}}}{2} \\
&= \frac{C-\beta}{2}\left(\frac{\beta}{2(1+\beta)} + \frac{\beta+2}{2(1+\beta)}\right) \\
&= \frac{C-\beta}{2}.
\end{aligned}$$

Thus,

$$\alpha_s \geq \frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{2}.$$

Now we prove that

$$\gamma_t \geq \frac{C-\beta}{C+\beta} \cdot \mu$$

for any $k \geq 0$ by induction.

The claim is correct for $t = 0$, by the choice of γ_0. Let us assume that $\gamma_t \geq \frac{C-\beta}{C+\beta} \cdot \mu$. This also implies that $\alpha_t \geq \frac{C-\beta}{2}$ by the previous argument. Then γ_{t+1} satisfies

$$(1+\beta)\gamma_{t+1} = (1-\alpha_t)\gamma_t + \alpha_t \mu.$$

Since $\alpha_t \leq 1$ and $\gamma_t \geq \frac{C-\beta}{C+\beta}\mu$, we have

$$(1-\alpha_t)\gamma_t + \alpha_t\mu \geq (1-\alpha_t)\frac{C-\beta}{C+\beta}\mu + \alpha_t\mu = (1-\alpha_t)\mu + \alpha_t\mu - (1-\alpha_t)\frac{2\beta}{C+\beta}\mu$$

$$=\mu + (\alpha_t - 1)\frac{2\beta}{C+\beta}\mu \geq \mu + \left(\frac{C-\beta}{2} - 1\right)\frac{2\beta}{C+\beta}\mu = \left(1 + \frac{(C-\beta)\beta}{C+\beta} - \frac{2\beta}{C+\beta}\right)\mu$$

$$=(1+\beta)\frac{C-\beta}{C+\beta}\mu.$$

Thus $\gamma_{t+1} \geq \frac{C-\beta}{C+\beta}\mu$ and the desired result holds.

For proving that $\gamma_t \geq \frac{\mu}{2}$, we only need to show that $\frac{C-\beta}{C+\beta} \geq \frac{1}{2}$, which is quite easy to see. This inequality can be written equivalently as

$$2C - 2\beta \geq C + \beta \Leftrightarrow C \geq 3\beta \Leftrightarrow \beta^2 + (1+\beta)\frac{\mu}{L} \geq 9\beta^2 \Leftrightarrow (1+\beta)\frac{\mu}{L} \geq 8\beta^2 = \frac{8}{25}\frac{\mu}{L}$$

Since $1 + \beta > 1$, the last inequality holds and the desired result follows. Thus, the sequence γ_t, created by $\bar{\gamma}_t$ as in the statement of the proposition, satisfies the requirements of step 9. ■

Proposition 6.8 leads us to a more concrete version of Algorithm 6.1 with a specific choice of hyperparameters:

Algorithm 6.2 Accelerated Gradient Descent for the Block Rayleigh Quotient

Initialize at $\mathcal{X}_0 = \mathcal{V}_0 \in \mathrm{Gr}(n,p)$ and choose shrinkage parameter $\beta = \frac{1}{5}\sqrt{\frac{\mu}{L}}$

Choose $\gamma_0 \geq \frac{\sqrt{\beta^2+(1+\beta)\frac{\mu}{L}}-\beta}{\sqrt{\beta^2+(1+\beta)\frac{\mu}{L}}+\beta} \cdot \mu$

for $k \geq 0$ **do**

$\beta_k = \underset{\eta\in[0,1]}{\mathrm{argmin}} \left\{ f(\mathrm{Exp}_{\mathcal{V}_t}(\eta \mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t))) \right\}$

$\mathcal{Y}_t = \mathrm{Exp}_{\mathcal{V}_t}(\beta_k \mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t))$

$4\alpha_t^2 = \frac{(1-\alpha_t)\gamma_t + \alpha_t \mu}{L}$

$\mathcal{X}_{t+1} = \mathrm{Exp}_{\mathcal{Y}_t}\left(-\frac{1}{L}\mathrm{grad} f(\mathcal{Y}_t)\right)$

$\bar{\gamma}_{t+1} = (1-\alpha_t)\gamma_t + \alpha_t \mu$

$\gamma_{t+1} = \frac{1}{1+\beta}\bar{\gamma}_{t+1}$

$\mathcal{V}_{t+1} = \mathrm{Exp}_{\mathcal{Y}_t}\left(\frac{(1-\alpha_t)\gamma_t}{\bar{\gamma}_{t+1}} \mathrm{Log}_{\mathcal{Y}_t}(\mathcal{V}_t) - \frac{2\alpha_t}{\bar{\gamma}_{t+1}} \mathrm{grad} f(\mathcal{Y}_t)\right)$

end for

From now on, we use Algorithm 6.2 as our standard algorithm for the rest of this section. Its convergence is analyzed in the next section.

6.5 Convergence

We are finally ready to complete the convergence analysis. We start with the following simple result.

Proposition 6.9 *The sequence $\mathcal{X}_t$ generated by Algorithm 6.2 satisfies*

$$f(\mathcal{X}_t) - f^* \leq \tau_t \left(f(\mathcal{X}_0) - f^* + \frac{\gamma_0}{2} \mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha) \right).$$

Proof We choose

$$\phi_0(\mathcal{X}) = f(\mathcal{X}_0) + \frac{\gamma_0}{2} \mathrm{dist}^2(\mathcal{X}_0, \mathcal{V}_\alpha)$$

as the beginning of the estimate sequence. Then $\phi_0^* = f(\mathcal{X}_0)$ and by construction of ϕ_t^* in Theorem 6.6, we get $f(\mathcal{X}_t) \leq \phi_t^*$. The result now follows by simply applying Proposition 6.2. ∎

Proposition 6.9 provides a worst-case upper bound for the sub-optimality of f and it only remains to estimate τ_t. Such an estimation can be easilty obtained by the proof of Proposition 6.8.

Proposition 6.10 *The sequence τ_t, defined recursively as $\tau_0 = 1$ and $\tau_{t+1} = (1-\alpha_t)\tau_t$, where α_t comes from Algorithm 6.2 starting from a point $\mathcal{X}_0$ such that*

$$\mathrm{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{1}{8}\sqrt{c_Q} \left(\frac{\delta}{L}\right)^{3/4},$$

is upper bounded as

$$\tau_t \leq \left(1 - \frac{2}{5}\sqrt{\frac{\mu}{L}}\right)^k.$$

Proof We have

$$\tau_t = \Pi_{i=0}^{t-1}(1 - \alpha_i)$$

and we only need to estimate a lower bound for α_i.

The proof of Proposition 6.8 provides a lower bound for α_i as

$$\alpha_i \geq \frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{2},$$

with $\beta = \frac{1}{5}\left(\frac{\mu}{L}\right)^{1/2}$. This is because we proved that $\gamma_i \geq \frac{C-\beta}{C+\beta} \cdot \mu$, for all i by induction and also proved that

$$\gamma_i \geq \frac{C-\beta}{C+\beta} \cdot \mu \Rightarrow \alpha_i \geq \frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{2}.$$

Taking into account the exact value for β, we can rewrite the lower bound for α_i as

$$\alpha_i \geq \frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{2} = \frac{1}{2}\sqrt{\frac{\mu}{L}}\left(\sqrt{\frac{1}{25} + 1 + \frac{4}{25}\sqrt{\frac{\mu}{L}}} - \frac{1}{5}\right) \geq \frac{2}{5}\sqrt{\frac{\mu}{L}}.$$

This provides the desired result. ■

We also rephrase the previous result in terms of iteration complexity:

Theorem 6.11 *Algorithm* 6.2 *starting from* $\mathcal{X}_0$ *satisfying*

$$\operatorname{dist}(\mathcal{X}_0, \mathcal{V}_\alpha) \leq \frac{1}{8}\sqrt{c_Q}\left(\frac{\delta}{L}\right)^{3/4},$$

computes an estimation $\mathcal{X}_T$ *of* $\mathcal{V}_\alpha$ *such that* $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$ *in at most*

$$T = \mathcal{O}\left(\sqrt{\frac{L}{\delta}}\log\frac{f(\mathcal{X}_0) - f^*}{\varepsilon\delta}\right)$$

many iterates.

Proof For $\operatorname{dist}(\mathcal{X}_T, \mathcal{V}_\alpha) \leq \epsilon$, it suffices to have

$$f(\mathcal{X}_T) - f^* \leq c_Q\epsilon^2\delta$$

by quadratic growth of f (Proposition 3.2). Using $(1-c)^t \leq \exp(-ct)$ for all $t \geq 0$ and $0 \leq c \leq 1$, Propositions 6.9 and 6.10 give that it suffices to choose t as the smallest integer such that

$$f(\mathcal{X}_t) - f^* \leq \exp\left(-\frac{2}{5}\sqrt{\frac{\mu}{L}}t\right)(f(\mathcal{X}_0) - f^*) \leq c_Q \epsilon^2 \delta.$$

Solving for t and substituting $\mu = 2c_Q\delta$, we get the required statement. ■

Remark Since the expression

$$\frac{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} - \beta}{\sqrt{\beta^2 + (1+\beta)\frac{\mu}{L}} + \beta} \cdot \mu$$

is strictly increasing with respect to μ, we can choose γ_0 by substituting μ with an over-approximation, for example γ:

$$\gamma_0 = \frac{\sqrt{\beta^2 + \beta + 1} - \beta}{\sqrt{\beta^2 + \beta + 1} + \beta} \cdot \gamma$$

6.6 Implementation details and computational cost of Algorithm 6.2

A naive implementation of Step 4 of Algorithm 6.2 can become quite costly as a simple binary search may need many function evaluations to reach β_k in a good accuracy, and as a result, many large matrix-vector multiplications. Fortunately, we can manipulate the expressions so that it suffices to do only two large matrix-vector multiplications. The idea for such a technique comes from [12].

Let $\mathcal{X}$ be a point on Grassmann and P a search direction. We consider the function

$$\mathcal{X}(\eta) = \mathrm{Exp}_{\mathcal{X}}(\eta P) = \mathrm{Span}(XV\cos(\eta\Sigma)V^T + U\sin(\eta\Sigma)V^T),$$

where X is a representative of $\mathcal{X}$ and $U\Sigma V^T$ a compact SVD of P. Here Σ is taken as a diagonal matrix and the functions sin and cos act only to its diagonal entries.

The value of f evaluated at $\mathcal{X}(\eta)$ is

$$\begin{aligned}
&f(\mathcal{X}(\eta)) \\
&= -\operatorname{Tr}((XV\cos(\eta\Sigma)V^T + U\sin(\eta\Sigma)V^T)^T A(XV\cos(\eta\Sigma)V^T + U\sin(\eta\Sigma)V^T)) \\
&= -\operatorname{Tr}(V\cos(\eta\Sigma)V^T X^T AXV\cos(\eta\Sigma)V^T) - \operatorname{Tr}(V\sin(\eta\Sigma)U^T AU\sin(\eta\Sigma)V^T) \\
&\quad -\operatorname{Tr}(V\cos(\eta\Sigma)V^T X^T AU\sin(\eta\Sigma)V^T) - \operatorname{Tr}(V\sin(\eta\Sigma)U^T AXV\cos(\eta\Sigma)V^T) \\
&= -\Sigma_{i=1}^k(\cos^2(\eta\Sigma_i)\alpha_i + 2\sin(\eta\Sigma_i)\cos(\eta\Sigma_i)\beta_i + \sin^2(\eta\Sigma_i)\gamma_i),
\end{aligned}$$

Matrix	n	κ	structure	fraction of nnz
FD3D	35 000	$7.0 \cdot 10^3$	real	$1.95 \cdot 10^{-4}$
ukerbe1	5 981	–	rank-deficient, binary	$4.39 \cdot 10^{-4}$
ACTIVSg70K	69 999	$2.9 \cdot 10^8$	real	$4.87 \cdot 10^{-5}$
boneS01	127 224	$4.2 \cdot 10^7$	real	$3.40 \cdot 10^{-4}$
audikw_1	943 695	–	rank-deficient, real	$8.7 \cdot 10^{-5}$

Table 6.1: Summary statistics of the tested matrices.

where

$$\alpha_i = (V^T X^T A X V)_{ii},\ \beta_i = (V^T X^T A U)_{ii} \text{ and } \gamma_i = (U^T A U)_{ii}.$$

Thus, for computing the steps 4-5 of Algorithm 6.2, we need to compute the matrix-vector products AV_t and AU, where U is the first matrix in the SVD of $\mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t)$. Then, we can execute binary search (or any accelerated version, including Newton's method) for calculating β_t without needing to compute any additional matrix-vector products with A. Moreover, these calculations are enough to provide immediately the product AY_t as

$$AY_t = (AV_t)V\cos(\beta_t\Sigma)V^T + (AU)\sin(\beta_t\Sigma)V^T,$$

where $U\Sigma V^T$ is the SVD of $\mathrm{Log}_{\mathcal{V}_t}(\mathcal{X}_t)$. Thus, for computing the gradient step (step 7) in Algorithm 6.2, we do not need to compute any new matrix-vector products as AY_t suffices for calculating $\mathrm{grad} f(\mathcal{Y}_t)$. Consequently, the cost of computing one iteration of Algorithm 6.2 is two matrix-vector products. This is more than gradient descent or conjugate gradients method [12] (that need only one matrix-vector product), but still reasonable as accelerated gradient descent typically features three kind of iterates ($\mathcal{X}_t$, $\mathcal{Y}_t$, $\mathcal{V}_t$).

6.7 Numerical experiments

We test the proposed method on a series of benchmark test matrices from the SuiteSparse Matrix Collection [32] used also by [96]. The main properties of the tested matrices, including their size n, condition number $\kappa = \lambda_1/\lambda_n$ and other structural properties, are summarized in Table 6.1. For each of the tested problems we also report the condition number

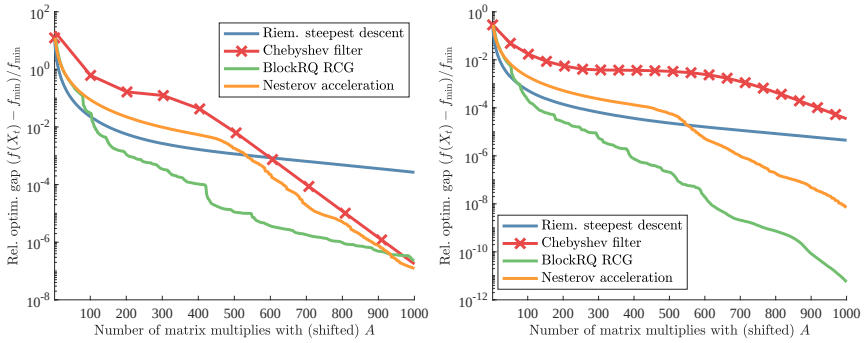
$$\kappa_{\mathrm{R}} = \frac{\lambda_1 - \lambda_n}{\lambda_k - \lambda_{k+1}} = \mathcal{O}\left(1/\delta\right), \tag{6.7.0.1}$$

of the Riemannian Hessian evaluated the dominant subspace with spectral gap δ [12, 14].

We compare the efficiency of the proposed Nesterov acceleration with three other methods: Riemannian gradient descent, Chebyshev filter in subspace iteration, and the Riemannian conjugate gradient for block Rayleigh quotient

Method	Mat. products per iter.	Required info.
Riem. gradient descent	1	–
Chebyshev filter	1	λ_{k+1}, λ_n
BlockRQ RCG	1	–
Nesterov acceleration	2	δ, κ

Table 6.2: Comparison of the number of matrix products with A required by each of the methods. *Riem. gradient descent and Chebyshev filter subspace method require an additional matrix product every s iterations where s corresponds to the degree of the filter polynomial.*



(a) Problem nb 1 ($k = 128$, min.) (b) Problem nb 2 ($k = 128$, max.)

Figure 6.1: The `FD3D` matrix.

(BlockRQ RCG). We precompute the eigenvalues using `eigs` command in MATLAB required to determine the optimal Chebyshev filter for the subspace iteration and the parameters in the Nesterov acceleration and BlockRQ RCG. The tested algorithms differ in the number of matrix-vector products that they require per iteration, which we summarize in Table 6.2.

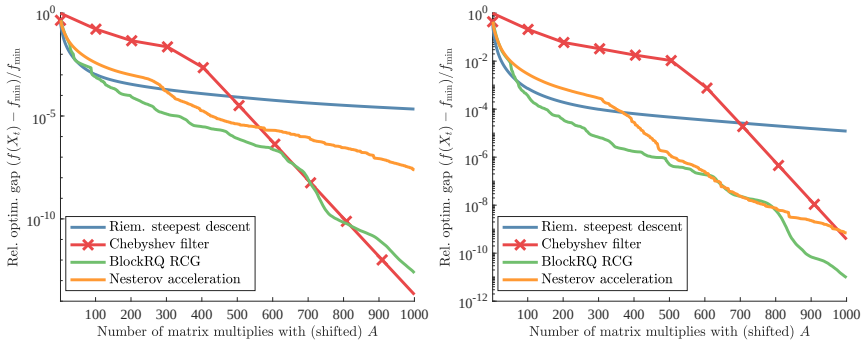
FD3D We generate a 3D finite difference Laplacian matrix corresponding to the uniform grid of size $35 \times 40 \times 25$ and zero Dirichlet boundary conditions, resulting in a matrix of size 35000.

The four problems with the finite difference matrix `FD3D` are summarized in Table 6.3. We experiment with computing the dominant subspace of dimension $k = 128$ and also with the minimization of the Rayleigh quotient.

problem nb (`FD3D`)	type	k	δ_k	κ_{R}	Cheb. degree
1	min	128	$8.3 \cdot 10^{-4}$	$1.4 \cdot 10^4$	100
2	max	128	$8.3 \cdot 10^{-4}$	$1.4 \cdot 10^4$	100

Table 6.3: Tested problems for the `FD3D` matrix ($n = 35000, \kappa = 7.0 \cdot 10^3$).

In Figure 6.1 we show the convergence plots tracking the number of matrix-vector products for problem nb 1 and 2. Overall, for both problems BlockRQ RCG outperforms the other methods, while Nesterov Acceleration matches the



(a) Problem nb 3 ($k = 64$) (b) Problem nb 4 ($k = 128$)

Figure 6.2: Comparison on `ukerbe1` test matrix on problem nb 3 and 4.

slow convergence of Riem. gradient descent for the first 250 iterations after which it starts converging at the proved rate $\mathcal{O}(1/\sqrt{\delta})$. The Chebyshev subspace iteration with polynomial of degree 100 is able to match the convergence rate of the Nesterov acceleration on problem nb 1 but not on problem nb 2.

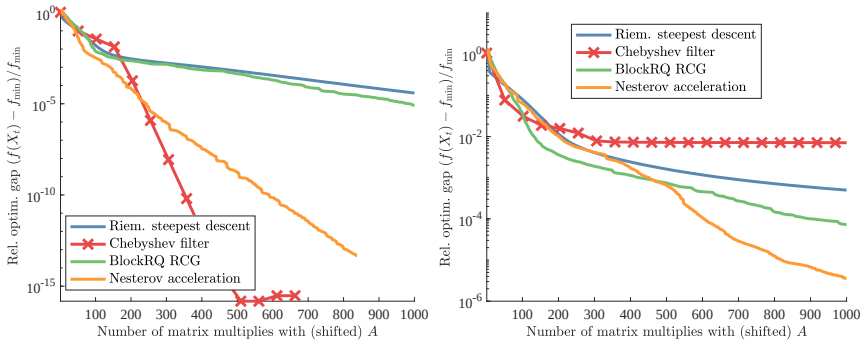
ukerbe1 The matrix comes from a locally refined non-uniform grid of a 2D finite element problem. Although the matrix is of a smaller size $n = 5981$ compared to the other tested matrices, it is a more challenging problem due to the non-uniform grid resulting in a very high condition number. The two tested problems are summarized in Table 6.4 and differ in the size of the subspace $k = 64$ and $k = 128$.

problem nb (`ukerbe1`)	type	k	δ_k	κ_{R}	Cheb. degree
3	max	64	$1.2 \cdot 10^{-3}$	$5.2 \cdot 10^{3}$	100
4	max	128	$9.4 \cdot 10^{-4}$	$6.7 \cdot 10^{3}$	100

Table 6.4: Tested problems for `ukerbe1` rank-deficient matrix ($n = 5981$).

In Figure 6.2 we see the performance of the methods on problem nb 3 and 4 for `ukerbe1` matrix. In both problems BlockRQ RCG and Chebyshev subspace iteration eventually outperform the Nesterov acceleration. We also observe an initial slow convergence of the Chebyshev iteration until the iteration 400 and 500 respectively.

ACTIVSg70K This large matrix models a synthetically generated power system grid. We experiment with subspace dimension $k = 32$ and $k = 64$ as described in Table 6.5.



(a) Problem nb 5 (max, $k = 32$) (b) Problem nb 6 (max, $k = 64$)

Figure 6.3: Comparison on `ACTIVSg70K` test matrix on problem nb 5 and 6.

problem nb (`ACTIVSg70K`)	type	k	δ_k	κ_{R}	Cheb. degree
5	max	32	$2.2 \cdot 10^2$	$1.2 \cdot 10^3$	50
6	max	64	$2.0 \cdot 10^2$	$1.3 \cdot 10^3$	50

Table 6.5: Tested problems for `ACTIVSg70K` matrix ($n = 69\,999, \kappa = 2.9 \cdot 10^8$).

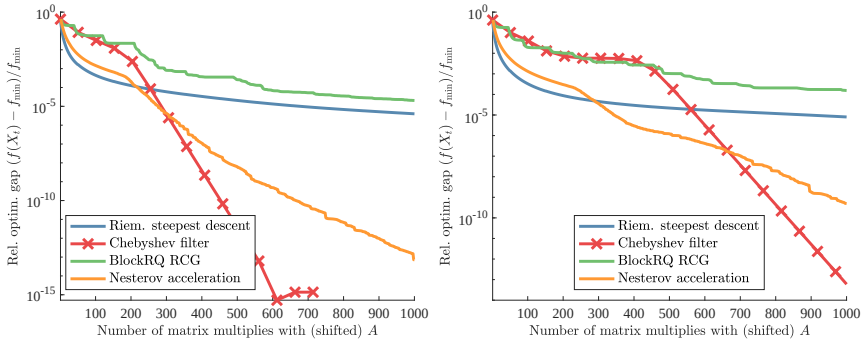
Figure 6.3 shows the convergence plots for `ACTIVSg70K`. We see that Nesterov acceleration outperforms the other methods on the problem with larger subspace dimension of $k = 64$ (which is harder). For the problem with smaller subspace of dimension $k = 32$, the Chebyshev iteration algorithm outperforms the other methods (while being the worst performing on $k = 64$), which reveals its sensitivity to choosing the correct degree of the filter polynomial.

boneS01 The second largest matrix we test is of size $n = 127224$ and comes from a finite element model studying the porous bone micro-architecture. The problem is challenging due to its large size and Riemannian condition number, see Table 6.6

problem nb (`boneS01`)	type	k	δ_k	κ_{R}	Cheb. degree
7	max	64	$2.4 \cdot 10^1$	$2.1 \cdot 10^3$	50
8	max	128	$1.3 \cdot 10^1$	$3.6 \cdot 10^3$	50

Table 6.6: Tested problems for `boneS01` matrix ($n = 127224, \kappa = 4.2 \cdot 10^7$).

Figure 6.4 shows the performance of the methods on `boneS01`. We see that the Chebyshev subspace iteration with degree 50, while having slower convergence at the beginning, outperforms the other methods. Nesterov acceleration is the second best performing and faster than Block RQ RCG and Riemannian gradient descent.



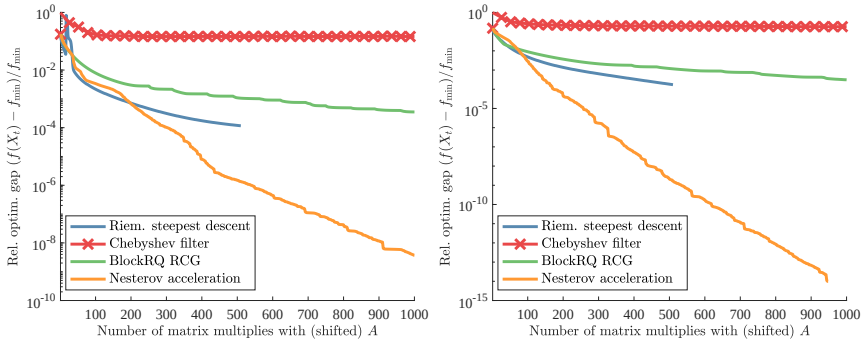
(a) Prob nb 7 (max $k = 64$) (b) Prob nb 8 (max $k = 128$)

Figure 6.4: Comparison on `boneS01` test matrix on problem 7 and 8.

audikw_1 The largest matrix of size $n = 943695$ we experiment with comes from finite elements problem modelling automotive crankshaft structure. The problem is challenging due to its large size and its large Riemannian condition number as can be seen in Table 6.7.

problem nb (`audikw_1`)	type	k	δ_k	κ_{R}	Cheb. degree
9	max	32	$4.3 \cdot 10^6$	$5.6 \cdot 10^3$	25
10	max	64	$1.9 \cdot 10^7$	$1.3 \cdot 10^3$	25

Table 6.7: Tested problems for `audikw_1` rank-deficient matrix ($n = 943\,695$).



(a) Prob nb 9 (max, $k = 64$) (b) Prob nb 10 (max, $k = 128$)

Figure 6.5: Comparison on `audikw_1` test matrix on problem nb 9 and 10.

In Figure 6.5 we see the convergence results for the largest tested matrix `audikw_1`. In this test, Nesterov acceleration clearly outperforms the other methods. The Chebyshev subspace iteration algorithm does not converge, which might be due to the wrong choice for the degree of the polynomial.

7 Polar decomposition

We now turn to the problem of computing the polar factor of a square matrix $C \in \mathbb{R}^{n\times n}$. This section follows the exposition of our work [13].

7.1 Introduction

As discussed in Section 1.2.2, computing a polar factor of a square matrix is equivalent to an orthogonal Procrustes problem. In this section, we reveal a convexity-like structure for this (generally non-convex) problem similar to the one for the symmetric eigenvalue problem in Section 2. Using this convexity-like structure, we analyze a Riemannian gradient descent algorithm in the orthogonal group for computing the polar factor of C. This algorithm is in general slow compared to the state-of-the-art and is presented only for theoretical purposes.

Although we have not yet developed a concrete state-of-the-art application of this theory, we believe it is highly likely to find use in noisy versions of polar decomposition, analogous to the theory developed for the symmetric eigenvalue problem. Potential applications include stochastic versions of the problem (where only an unbiased estimate of the matrix C is available, typically requiring the use of stochastic algorithms) and robust formulations of polar decomposition. The latter can be to solve the optimization problem

$$\min_{X\in\mathbb{O}(n)} \max_{C\in\mathbb{R}^{n\times n}} \left(-\operatorname{Tr}(CX) - \beta \sum_{i=1}^{s} \|C - C_i\|^2 \right), \tag{7.1.0.1}$$

where $\{C_i\}_{i=1}^s$ is a set of independent observations for C and $\beta > 0$ is a regularizer. To the best of our knowledge, traditional linear algebra techniques cannot be applied to such problem. A more viable approach would be min-max optimization (for instance gradient descent-ascent), for which our theory could be of value.

7.2 Convexity-like properties of orthogonal Procrustes

We investigate now thoroughly the orthogonal Procrustes problem. This problem concerns with finding orthogonal matrices X_1 and X_2 that best fit two other matrices $A, B \in \mathbb{R}^{m\times n}$:

$$\min_{X_1,X_2\in\mathbb{O}(n)} \|AX_1 - BX_2\|_F^2.$$

Since this problem is invariant under simultaneous right multiplication of X_1 and X_2 with an orthogonal matrix $Q \in \mathbb{R}^{n\times n}$, we can fix X_2 to be identity and target only the matrix $X_1 \rightsquigarrow X$:

$$\min_{X\in\mathbb{O}(n)} \|AX - B\|^2.$$

This problem can be written equivalently as

$$\min_{X \in \mathbb{O}(n)} -\operatorname{Tr}(CX) =: f(X), \tag{7.2.0.1}$$

where

$$C := B^T A.$$

In addition, this problem has a global solution and can be found in closed form [98]: if $C = U\Sigma V^T$ is an SVD of C, then a global solution is $X^* = VU^T$. The minimum $f^* := f(X^*)$ is the opposite of the sum of the singular values of C.

We will use this structure to prove a quasi-convexity property for the function

$$f(X) = -\operatorname{Tr}(CX)$$

around X^*.

It is well known that the solution of the problem is *unique* if and only if all the singular values of C are strictly positive, i.e. if and only if C is invertible.

Riemannian gradient: To compute the Riemannian gradient of f, we just need to project the Euclidean gradient $\nabla f(X) = -C^T$ onto the tangent space $T_X\mathbb{O}(n)$. This results to

$$\operatorname{grad} f(X) = P_X(-C^T) = -X\operatorname{skew}(X^T C^T). \tag{7.2.0.2}$$

Riemannian Hessian: For a function f defined in the orthogonal group, we have (see [20])

$$D\operatorname{grad} f(X)[\dot{X}] = \dot{X}\operatorname{skew}(X^T\nabla f(X)) + X\operatorname{skew}(\dot{X}^T\nabla f(X) + \dot{X}^T\nabla^2 f(X)[\dot{X}]),$$

where $\dot{X} = X\Omega$ is an arbitrary tangent vector. In our case, $\nabla f(X) = -C^T$ and $\nabla^2 f(X) = 0$, thus

$$\operatorname{Hess} f(X)[\dot{X}] = -\dot{X}\operatorname{skew}(X^T C^T) - X\operatorname{skew}(\dot{X}^T C^T). \tag{7.2.0.3}$$

We now show a weak-quasi convexity property for f, similar to Proposition 2.6.

Proposition 7.1 (Geodesic weak-quasi convexity) *Let $X^* \in \mathbb{O}(n)$ a global optimum of the function $f : \mathbb{O}(n) \to \mathbb{R}$. Let also $X \in \mathbb{O}(n)$ such that the eigenvalues e^{ir} of $X^T X^*$ are such that $r \in (-\pi, \pi)$. If $|r|_{\max}$ denotes the largest possible rotation induced by $X^T X^*$ in absolute value, then*

$$\langle \operatorname{grad} f(X), -\operatorname{Log}_X(X^*)\rangle \geq \frac{1}{2}(1 + \cos(|r|_{\max}))(f(X) - f^*).$$

Proof The Riemannian gradient of f is given in equation (7.2.0.2). It remains to compute a convenient expression for the Riemannian logarithm. According to equation (1.3.1.10), the Riemannian logarithm is given as

$$\mathrm{Log}_X(X^*) = X \log_m(X^T X^*).$$

As in the introduction (Section 1.2.2), we use the canonical form of the orthogonal matrix $X^T X^*$:

$$X^T X^* = PDP^T.$$

Since the matrix logarithm is invariant under conjugate action, we have

$$\log_m(X^T X^*) = P \log_m(D) P^T$$

and $\log_m(D)$ is again a block diagonal matrix, with blocks being the logarithms of the blocks of D: when D has a diagonal entry equal to 1, $\log_m(D)$ has a diagonal entry equal to 0 and when D features a 2×2 block, which is a rotation of angle r, $\log_m(D)$ features the block $\begin{bmatrix} 0 & -r \\ r & 0. \end{bmatrix}$

Similarly, the skew-symmetric part of $X^T X^*$ satisfies

$$\mathrm{skew}(X^T X^*) = P \mathrm{skew}(D) P^T,$$

where $\mathrm{skew}(D)$ is again block diagonal and has a 0 diagonal entry when D has a 1 diagonal entry, while it has a block $\begin{bmatrix} 0 & -\sin r \\ \sin r & 0 \end{bmatrix}$ when D features a 2×2 rotation of angle r. Thus, it holds in general that

$$\log_m(D) = \mathrm{skew}(D) \frac{\phi}{\sin \phi},$$

where $\phi = (r_1, \ldots, r_n)$ is a vector capturing all the rotations induced by the orthogonal matrix $X^T X^*$. If $r = 0$, i.e. corresponds to a diagonal entry equal to 1, then it appears only once in ϕ, while if $r \in (-\pi, \pi) \setminus \{0\}$ it appears as a couple with $-r$.
$\phi / \sin \phi$ is a diagonal matrix with diagonal elements $r_j / \sin r_j$. This convention is made for ease of notation.

Given that, we can write

$$\mathrm{Log}_X(X^*) = X \log_m(X^T X^*) = XP \log_m(D) P^T = XP \mathrm{skew}(D) \frac{\phi}{\sin \phi} P^T$$

$$= XP \mathrm{skew}(D) P^T P \frac{\phi}{\sin \phi} P^T = P_X(X^*) P \frac{\phi}{\sin \phi} P^T.$$

Now we can finally deal with the desired inequality:

$$\langle \mathrm{grad} f(X), -\mathrm{Log}_X(X^*) \rangle = \left\langle P_X(C^T), P_X(X^*) P \frac{\phi}{\sin \phi} P^T \right\rangle =$$

$$\left\langle X \mathrm{skew}(X^T C^T), X^* P \frac{\phi}{\sin \phi} P^T \right\rangle = \mathrm{Tr} \left(P \frac{\phi}{\sin \phi} P^T X^{*T} X \mathrm{skew}(X^T C^T) \right).$$

We pause to deal with the term $X^{*T}X\text{skew}(X^TC^T)$:

$$X^{*T}X\text{skew}(X^TC^T) = X^{*T}X\frac{X^TC^T - CX}{2} = \frac{X^{*T}C^T - X^{*T}XCX}{2}.$$

Remember that if $X^* = VU^T$, then $C = U\Sigma V^T$ is an SVD of C. Thus

$$X^{*T}C^T = U\Sigma U^T$$

and

$$X^{*T}XCX = PD^TP^TU\Sigma V^TVU^TPD^TP^T = PD^TP^TU\Sigma U^TPD^TP^T.$$

Plugging this expression in, we get

$$\begin{aligned}
&2\langle \text{grad}f(X), -\text{Log}_X(X^*)\rangle = \text{Tr}\left(P\frac{\phi}{\sin\phi}P^TX^{*T}X\text{skew}(X^TC^T)\right)\\
&= \text{Tr}\left(P\frac{\phi}{\sin\phi}P^T(U\Sigma U^T - PD^TP^TU\Sigma U^TPD^TP^T)\right)\\
&= \text{Tr}\left(\frac{\phi}{\sin\phi}(P^TU\Sigma U^TP - D^TP^TU\Sigma U^TPD^T)\right)\\
&= \text{Tr}\left(\frac{\phi}{\sin\phi}(P^TU\Sigma U^TP)\right) - \text{Tr}\left(\frac{\phi}{\sin\phi}(D^TP^TU\Sigma U^TPD^T)\right)\\
&= \text{Tr}\left(\left(\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T\right)P^TU\Sigma U^TP\right).
\end{aligned}$$

It suffices to show that

$$\text{Tr}\left(\left(\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T\right)P^TU\Sigma U^TP\right) \geq (1+\cos(|r|_{\max}))(f(X) - f^*) =$$

$$\underbrace{(1+\cos(|r|_{\max}))}_{:=c}\left(\underbrace{\text{Tr}(P^TU\Sigma U^TP)}_{-f^*} - \underbrace{\text{Tr}(D^TP^TU\Sigma U^TP)}_{-f(X)}\right).$$

This holds if

$$\text{Tr}\left(\left(\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T + c(D^T - I)\right)\underbrace{P^TU\Sigma U^TP}_{:=A}\right) \geq 0.$$

Notice that the matrix A is symmetric and positive semi-definite.

D^T is a matrix with diagonal entries equal to 1 and 2×2 diagonal blocks of the form $\begin{bmatrix}\cos r & \sin r\\ -\sin r & \cos r\end{bmatrix}$, which essentially correspond to rotations with

$-r$. Multiplying with the diagonal matrix $\phi/\sin\phi$ from the right, keeps the 1 diagonal entries of D^T unchanged, while it transforms the 2×2 diagonal blocks to $\begin{bmatrix} r/\tan r & r \\ -r & r/\tan r \end{bmatrix}$. The matrix $D^T\frac{\phi}{\sin\phi}D^T$ still keeps 1 in the entries that correspond to $r=0$ and has 2×2 diagonal blocks associated with $r\in(-\pi,\pi)\setminus\{0\}$ that are $\begin{bmatrix} \frac{r}{\tan r}\cos r - r\sin r & \frac{r}{\tan r}\sin r + r\cos r \\ -r\cos r - \frac{r}{\tan r}\sin r & \frac{r}{\tan r}\cos r - r\sin r \end{bmatrix}$.

The matrix $\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T + c(D^T - I)$ has 1 in the diagonal entries that D^T has 1 ($r=0$) and has 2×2 diagonal blocks that correspond to rotations with $r\in(-\pi,\pi)\setminus\{0\}$, which are

$$\begin{bmatrix} \frac{r}{\sin r} - \frac{r}{\tan r}\cos r + r\sin r + c(\cos r - 1) & -\frac{r}{\tan r}\sin r - r\cos r + c\sin r \\ \frac{r}{\tan r}\sin r + r\cos r - c\sin r & \frac{r}{\sin r} - \frac{r}{\tan r}\cos r + r\sin r + c(\cos r - 1) \end{bmatrix}.$$

Notice that this last 2×2 matrix is of the form $\begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix}$.

The expression $\mathrm{Tr}\left(\left(\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T + c(D^T - I)\right)A\right)$ that we want to prove nonnegative is the sum of the traces of the product of the diagonal entries of $\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T + c(D^T - I)$ that correspond to $r=0$ (i.e. 1) with the corresponding diagonal entries of A and the 2×2 diagonal blocks of $\frac{\phi}{\sin\phi} - D^T\frac{\phi}{\sin\phi}D^T + c(D^T - I)$ with the corresponding 2×2 diagonal blocks of A. In the first case we get back the diagonal entries of A (which are nonnegative) and in the second case we have the product of a matrix of the form $\begin{bmatrix} \alpha & \beta \\ -\beta & \alpha \end{bmatrix}$ with one of the form $\begin{bmatrix} s & t \\ t & k \end{bmatrix}$, since A is symmetric. The diagonal entries of this product (which are the ones that contribute in the trace) are $\alpha s + \beta t$ and $-\beta t + \alpha k$. Their sum is $\alpha(s+t)$, thus it suffices to show that this expression is nonnegative, i.e. that α is nonnegative since s and t are nonnegative as diagonal entries of the positive semi-definite matrix A.

Remember that α has been taken as

$$\begin{aligned} \alpha :&= \frac{r}{\sin r} - \frac{r}{\tan r}\cos r + r\sin r + \underbrace{(1+\cos(|r|_{\max}))}_{c}(\cos r - 1) \\ &\geq \frac{r}{\sin r} - \frac{r}{\tan r}\cos r + r\sin r + (1+\cos r)(\cos r - 1), \end{aligned}$$

since $r\leq |r|_{\max}$ and $\cos r - 1\leq 0$. The last lower bound for α turns out to be positive for all $r\in(-\pi,\pi)$, thus our proof is complete.

■

We now examine a property for f known as quadratic growth. This property gives a non-trivial inequality only in the case that the Procrustes problem has a unique solution (i.e. if and only if C is non-singular). This is similar to Proposition 2.4.

Proposition 7.2 (Quadratic growth) *Let $X^* \in \mathbb{O}(n)$ to be a global minimizer for f and $X \in \mathbb{O}(n)$ in the same connected component. Then f satisfies*

$$f(X) - f^* \geq \frac{2\sigma_{min}(C)}{\pi^2}\mathrm{dist}^2(X, X^*),$$

where $\sigma_{\min}(C)$ is the smallest singular value of C.

Proof Recall that if $C = U\Sigma V^T$ is an SVD of C, then $X^* = VU^T$ is a global minimizer. Consider again the canonical form of the orthogonal matrix $X^T X^*$:

$$X^T X^* = PDP^T.$$

Then, we have

$$\begin{aligned} f(X) - f^* &= -\operatorname{Tr}(CX) + \operatorname{Tr}(CX^*) = \operatorname{Tr}(P^T U\Sigma U^T P) - \operatorname{Tr}(U\Sigma U^T P D^T P^T) \\ &= \operatorname{Tr}((I - D^T)\underbrace{P^T U\Sigma U^T P}_{\text{pos. semi-definite}}) \end{aligned}$$

Let us denote again $A := P^T U\Sigma U^T P$, which is symmetric and positive semi-definite. The matrix $I - D^T$ has diagonal entries equal to 0 for rotations $r = 0$, diagonal entries equal to 2 for $r = \pi$ and 2×2 diagonal blocks of the form $\begin{bmatrix} 1-\cos r & \sin r \\ -\sin r & 1-\cos r \end{bmatrix}$ for rotations with angle $r \in (-\pi, \pi)$ not 0. Thus, the diagonal entries of the product $(I - D^T)P^T U\Sigma U^T P$ are either 0 for entries that correspond to no rotation, i.e. $(1 - \cos r)A_{ii}$, or the diagonal entries of a product of the form

$$\begin{bmatrix} 1-\cos r & \sin r \\ -\sin r & 1-\cos r \end{bmatrix} \begin{bmatrix} s & t \\ t & k \end{bmatrix}.$$

These are $(1 - \cos r)s + \sin rt$ and $-\sin rt + (1 - \cos r)k$. Since summing them makes the terms $\sin rt$ to cancel out, we get

$$\operatorname{Tr}((I - D^T)P^T U\Sigma U^T P) = \operatorname{Tr}((I - \cos\phi)P^T U\Sigma U^T P),$$

where $\phi = (r_1, \ldots, r_n)$ a vector capturing all the rotations between X and X^*. If $r_j = 0$ or π, then it appears only once, if $r_j \neq 0, \pi$ it appears coupled with its opposite $-r$. Notice that

$$\|\phi\| = \mathrm{dist}(X, X^*).$$

Since for all r it holds $r \in (-\pi, \pi]$, we have

$$1 - \cos r \geq \frac{2}{\pi^2} r^2.$$

By basic properties of the trace, we have

$$\mathrm{Tr}((I - \cos\phi) P^T U \Sigma U^T P) \geq \lambda_{\min}(P^T U \Sigma U^T P)\, \mathrm{Tr}(I - \cos\phi) \geq \frac{2\sigma_{\min}(C)}{\pi^2} \|\phi\|^2.$$

The last inequality completes the proof.

■

We can combine Propositions 7.1 and 7.2 to a more compact form, which we call weak-quasi-strong convexity (WQSC). This is similar to Theorem 2.7.

Interestingly, the role of a strong convexity constant μ is played by a multiple of $\sigma_{\min}(C)$. That is to say, the further away from being singular C is, the stronger this property becomes. If C is singular, the derived inequality reduces to weak-quasi convexity (Proposition 7.1, but with slightly weaker parameters).

Proposition 7.3 (Weak-quasi-strong convexity) *For any X satisfying the properties of Propositions* 7.1, 7.2*, f satisfies the following inequality:*

$$f(X) - f^* \leq \frac{1}{a(X)} \langle \mathrm{grad} f(X), -\mathrm{Log}_X(X^*) \rangle - \frac{\mu}{2} \mathrm{dist}^2(X, X^*)$$

with $a(X) := \frac{1+\cos(|r|_{max})}{4}$ and $\mu := \frac{4\sigma_{min}(C)}{\pi^2}$. $|r|_{max} < \pi$ is the largest rotation in absolute value induced by the orthogonal matrix $X^T X^*$.

Proof For the specific choices of $a(X)$ and μ, we have

$$\frac{\mu}{2} \mathrm{dist}^2(X, X^*) \leq f(X) - f^* \leq \frac{1}{2a(X)} \langle \mathrm{grad} f(X), -\mathrm{Log}_X(X^*) \rangle.$$

The left inequality is derived by Proposition 7.2 and the right one by Proposition 7.1.

Now, again by Proposition 7.1, we have

$$\begin{aligned} f(X) - f^* &\leq \frac{1}{2a(X)} \langle \mathrm{grad} f(X), -\mathrm{Log}_X(X^*) \rangle + \frac{\mu}{2} \mathrm{dist}^2(X, X^*) - \frac{\mu}{2} \mathrm{dist}^2(X, X^*) \\ &\leq \frac{1}{a(X)} \langle \mathrm{grad} f(X), -\mathrm{Log}_X(X^*) \rangle - \frac{\mu}{2} \mathrm{dist}^2(X, X^*) \end{aligned}$$

by substituting the previous inequality. ■

We close this exploration around a convexity-like structure for f, by examining its smoothness properties.

Proposition 7.4 (Smoothness) *f is geodesically $\sigma_{\max}(C)-$smooth.*

Proof It suffices to show that the eigenvalues of the Riemannian Hessian at X are upper bounded in absolute value by $\sigma_{\max}(C)$ for all X. For our computations, we follow the exposition of [20]:

$$\langle \dot{X}, \mathrm{Hess} f(X)[\dot{X}] \rangle = \mathrm{Tr}(\dot{X}^T \dot{X} \mathrm{skew}(-X^T C^T)) + \mathrm{Tr}(\dot{X}^T X \mathrm{skew}(-\dot{X}^T C^T)).$$

The first term is 0 as the trace of the product of a symmetric and skew-symmetric matrix. The second term becomes

$$\mathrm{Tr}(\dot{X}^T X \mathrm{skew}(-\dot{X}^T C^T)) = \frac{1}{2} \mathrm{Tr}(\dot{X}^T X C \dot{X} - \dot{X}^T X \dot{X}^T C^T).$$

Substituting $\dot{X}^T X = \Omega^T$, we get

$$\frac{1}{2} \mathrm{Tr}(\dot{X}^T X C X - \dot{X}^T X \dot{X}^T C^T) = \frac{1}{2} \mathrm{Tr}(\Omega^T C \dot{X} - \Omega^T \dot{X}^T C^T) =$$
$$\frac{1}{2} \mathrm{Tr}(\Omega^T C \dot{X} + \Omega \dot{X}^T C^T) = \mathrm{Tr}(\Omega^T C \dot{X}) = \mathrm{Tr}(\Omega^T C X \Omega) = \mathrm{Tr}(C X \Omega \Omega^T).$$

The last expression features the trace of the product of the matrix CX with the symmetric and positive semi-definite matrix $\Omega\Omega^T$. By basic facts in linear algebra, we can upper bound the absolute value of this expression by $\sigma_{\max}(CX)\,\mathrm{Tr}(\Omega\Omega^T)$. Since X is orthogonal, we have that $\sigma_{\max}(CX) = \sigma_{\max}(C)$. Also $\mathrm{Tr}(\Omega\Omega^T) = \mathrm{Tr}(\dot{X}\dot{X}^T) = \|\dot{X}\|^2$. Putting it all together, we get

$$|\langle \dot{X}, \mathrm{Hess} f(X)[\dot{X}] \rangle| \leq \sigma_{\max}(C) \|\dot{X}\|^2$$

and the desired result follows. ■

As it is customary, we denote

$$L := \sigma_{\max}(C).$$

We conclude this section with a small technical lemma that allows us to show that gradient descent with a properly chosen step size is well-defined in the sense that the direction used for update belongs in the injectivity domain (1.3.1.8).

Lemma 7.5 *The Riemannian gradient of f evaluated at X is of the form $X\Omega$, for a skew-symmetric matrix Ω with*

$$\|\Omega\|_2 \leq \sigma_{\max}(C).$$

Proof The Riemannian gradient of f at X is

$$\mathrm{grad} f(X) = X \mathrm{skew}(X^T C^T),$$

thus Ω is taken as $\mathrm{skew}(X^TC^T)$. By the sub-additivity of the spectral norm and its invariance under multiplication with orthogonal matrices, we have

$$\|\Omega\|_2 = \|\mathrm{skew}(X^TC^T)\|_2 \leq \frac{\|X^TC^T\|_2 + \|CX\|_2}{2} = \frac{\|C^T\|_2 + \|C\|_2}{2}.$$

This gives the desired result. ■

7.3 Convergence of Riemannian gradient descent

Riemannian gradient descent applied to a function $f : \mathbb{O}(n) \to \mathbb{R}$ reads as

$$X_{t+1} = \mathrm{Exp}_{X_t}(-\eta_t \mathrm{grad} f(X_t)), \tag{7.3.0.1}$$

with $\eta_t > 0$ being the step size.

The results of Section 7.2 guarantee a local (non-asymptotic) linear convergence rate for Riemannian gradient descent on f in the case that C is invertible, if ran with a properly chosen step size and the initial guess X_0 is sufficiently close to the optimum. We again emphasize that this is not a practical algorithm and is presented for theoretical purposes and to match the discussion of Section 2.

Proposition 7.6 *Let X_t and X^* be such that the largest rotation $|r|_{\max}$ induced by the orthogonal matrix $X_t^TX^*$ satisfies $|r|_{\max} < \pi$. Then, iteration* (7.3.0.1) *with $0 \leq \eta_t \leq a(X_t)/L$ satisfies*

$$\mathrm{dist}^2(X_{t+1}, X^*) \leq \left(1 - \frac{4}{\pi^2}\sigma_{\min}(C)a(X_t)\eta_t\right)\mathrm{dist}^2(X_t, X^*),$$

with $a(X_t)$ defined as in Proposition 7.3.

Proof We start by showing that iteration (7.3.0.1) is well-defined. By the assumption $|r|_{\max} < \pi$, we get that $0 < a(X_t) = \frac{1+\cos(|r|_{\max}))}{4} \leq \frac{1}{2}$. By Lemma 7.5, the tangent vector $\eta_t \mathrm{grad} f(X_t)$ that is used to update iteration (7.3.0.1) can be written as $X\Omega$, with $\|\Omega\|_2 \leq \eta_t\sigma_{max}(C)$. By the definition of η_t, we have that

$$\|\Omega\|_2 \leq \frac{a(X_t)}{L}\sigma_{\max}(C) = \frac{a(X_t)}{\sigma_{\max}(C)}\sigma_{\max}(C) \leq \frac{1}{2}.$$

Thus, $\eta_t \mathrm{grad} f(X_t)$ is inside the injectivity domain (1.3.1.8) and, as a consequence, iteration (7.3.0.1) is well-defined.

We can now apply Proposition 1.15 to obtain

$$\begin{aligned}\mathrm{dist}^2(X_{t+1}, X^*) &\leq \| - \eta_t \mathrm{grad} f(X_t) - \mathrm{Log}_{X_t}(X^*)\|^2 \\ &= \eta_t^2\|\mathrm{grad} f(X_t)\|^2 + \mathrm{dist}^2(X_t, X^*) + 2\eta_t\,\sigma \end{aligned} \tag{7.3.0.2}$$

with

$$\sigma := \langle \mathrm{grad} f(X_t), \mathrm{Log}_{X_t}(X^*) \rangle.$$

Propositions 7.3 and 7.4 (see also Proposition (1.21)) give

$$\begin{aligned} \frac{\sigma}{a(X_t)} &\leq f^* - f(X_t) - \frac{2\sigma_{\min}(C)}{\pi^2} \mathrm{dist}^2(X_t, X^*) \\ &\leq -\frac{1}{2L} \|\mathrm{grad} f(X_t)\|^2 - \frac{2\sigma_{\min}(C)}{\pi^2} \mathrm{dist}^2(X_t, X^*). \end{aligned}$$

Multiplying by $2a(X_t)\,\eta_t$ and using $\eta_t \leq a(X_t)/L$, we get

$$\begin{aligned} 2\eta_t \sigma &\leq -\frac{a(X_t)\eta_t}{L} \|\mathrm{grad} f(X_t)\|^2 - \frac{4\sigma_{\min}(C)}{\pi^2} a(X_t)\eta_t \mathrm{dist}^2(X_t, X^*) \\ &\leq -\eta_t^2 \|\mathrm{grad} f(X_t)\|^2 - \frac{4\sigma_{\min}(C)}{\pi^2} a(X_t)\eta_t \mathrm{dist}^2(X_t, X^*). \end{aligned}$$

Substituting into equation (7.3.0.2), we obtain the desired result. ■

Theorem 7.7 (Convergence of RGD for the procrustes problem) *Let C be invertible ($\sigma_{\min}(C) > 0$) and X^* the (unique) minimizer of f. Then, Riemannian gradient descent* (7.3.0.1) *in the orthogonal group, starting by a point $X_0 \in \mathbb{O}(n)$ such that*

$$\mathrm{dist}(X_0, X^*) < \pi,$$

and ran with fixed step size

$$\eta_t \equiv \eta \leq \frac{1 + \cos(\mathrm{dist}(X_0, X^*))}{4\sigma_{\max}(C)},$$

produces iterates X_t that satisfy

$$\mathrm{dist}^2(X_t, X^*) \leq \left(1 - \frac{1}{\pi^2}(1 + \cos(\mathrm{dist}(X_0, X^*)))\sigma_{\min}(C)\eta\right)^t \mathrm{dist}^2(X_0, X^*).$$

Proof We do the proof by induction.

For $t = 0$, the inequality is trivially true.

We now assume that the inequality is true for t and we wish to show that it is true also for $t + 1$.

Since $\mathrm{dist}(X_t, X^*) \leq \mathrm{dist}(X_0, X^*)$, we also get that the largest possible rotation $|r(X_t, X^*)|_{\max}$ induced by $X_t^T X^*$ satisfies

$$|r(X_t, X^*)|_{\max} \leq \sqrt{\sum_{i=1}^{n} r_i(X_t, X^*)^2} = \mathrm{dist}(X_t, X^*) \leq \mathrm{dist}(X_0, X^*),$$

where $r_i(X_t, X^*)$ are the rotations induced by the matrix $X_t^T X^*$. The equality in the previous derivation comes from equation (1.3.1.11).

By the definition of $a(X_t)$ in Proposition (7.3), we have

$$a(X_t) = \frac{1 + \cos(|r(X_t, X^*)|_{\max})}{4} \geq \frac{1 + \cos(\text{dist}(X_0, X^*))}{4},$$

thus $\eta \leq a(X_t)/L$.

Since η satisfies the previous bound, the outcome of Proposition 7.6 holds, and combining it with the induction hypothesis, we get

$$\begin{aligned}
&\text{dist}^2(X_{t+1}, X^*) \leq \left(1 - \frac{4}{\pi^2}\sigma_{\min}(C)a(X_t)\eta\right) \text{dist}^2(X_t, X^*) \leq \\
&\left(1 - \frac{1}{\pi^2}(1 + \cos(\text{dist}(X_0, X^*)))\sigma_{\min}(C)\eta\right) \text{dist}^2(X_t, X^*) \leq \\
&\left(1 - \frac{1}{\pi^2}(1 + \cos(\text{dist}(X_0, X^*)))\sigma_{\min}(C)\eta\right)^{t+1} \text{dist}^2(X_0, X^*).
\end{aligned}$$

This concludes the induction.

■

Remark 7.1 *If C is singular, then the previous theorem only states that the distances of the iterates of gradient descent to the set of optima do not increase. In that case we can still prove an algebraic convergence rate for the function values of Riemannian gradient descent based only on weak-quasi convexity.*

Remark 7.2 *The assumption $\text{dist}(X_0, X^*) < \pi$ allows to bound globally $|r(X_t, X^*)|_{\max}$ from above by $\text{dist}(X_0, X^*)$ and as a result keep the quantity $1+\cos(|r(X_t, X^*)|_{\max})$ far away from 0 over the course of gradient descent. Intuitively, it does not allow the algorithm to go too close to non-optimal critical points. Gradient descent would not stick to non-optimal critical points, but it would probably slow down a lot.*

We close this section by showing an algebraic convergence rate for gradient descent that covers also the case that C is singular.

Theorem 7.8 *Gradient descent applied to f for any square non-zero matrix C, starting from $X_0 \in \mathbb{O}(n)$ such that*

$$\text{dist}(X_0, X^*) < \pi$$

and with fixed step size

$$\eta \leq \frac{1 + \cos(\text{dist}(X_0, X^*))}{4\sigma_{\max}(C)},$$

produces iterates X_t that satisfy

$$f(X_t) - f^* \leq \frac{2L + \frac{1}{\eta}}{(1 + \cos(\mathrm{dist}(X_0, X^*)))t + 4} \mathrm{dist}^2(X_0, X^*) = \mathcal{O}\left(\frac{1}{t}\right).$$

Proof Since we still satisfy all the hypotheses of Theorem 7.7, we know that for all $t \geq 0$ it holds $\mathrm{dist}(X_t, X^*) \leq \mathrm{dist}(X_0, X^*) < \pi$. This implies that

$$a(X_t) \geq \frac{1 + \cos(\mathrm{dist}(X_0, X^*))}{4} > 0,$$

which implies that the function f is weakly-quasi-convex (Proposition 7.1) at every X_t such that:

$$\langle \mathrm{grad} f(X_t), -\mathrm{Log}_X(X^*) \rangle \geq \frac{1}{2}(1 + \cos(\mathrm{dist}(X_0, X^*)))(f(X_t) - f^*).$$

Denoting $C_0 := \frac{1+\cos(\mathrm{dist}(X_0,X^*))}{4}$ and $\Delta_t := f(X_t) - f^*$, we can write

$$2C_0\Delta_t \leq \langle \mathrm{grad} f(X_t), -\mathrm{Log}_{X_t}(X^*) \rangle. \tag{7.3.0.3}$$

Similarly to the proof of Proposition 7.6, by the hypothesis on the step size η_t, Lemma 7.5 shows that $-\eta_t X_{t+1}$ is in the injectivity domain of exp at X_t. Hence, by the definition of Riemannian gradient descent, we have

$$\mathrm{Log}_{X_t}(X_{t+1}) = -\eta \mathrm{grad} f(X_t). \tag{7.3.0.4}$$

In addition, the smoothness property of f (Proposition 7.4) gives

$$\Delta_{t+1} - \Delta_t \leq \langle \mathrm{grad} f(X_t), \mathrm{Log}_{X_t}(X_{t+1}) \rangle + \frac{L}{2} \mathrm{dist}^2(X_t, X_{t+1}).$$

Substituting (7.3.0.4), we obtain

$$\Delta_{t+1} - \Delta_t \leq \left(-\eta + \frac{L}{2}\eta^2\right) \|\mathrm{grad} f(X_t)\|^2 \leq 0. \tag{7.3.0.5}$$

By Proposition 1.15, we have

$$\mathrm{dist}^2(X_{t+1}, X^*) \leq \mathrm{dist}^2(X_t, X_{t+1}) + \mathrm{dist}^2(X_t, X^*) - 2\langle \mathrm{Log}_{X_t}(X_{t+1}), \mathrm{Log}_{X_t}(X^*) \rangle.$$

Substituting (7.3.0.4) into the above and rearranging terms gives

$$2\eta \langle \mathrm{grad} f(X_t), -\mathrm{Log}_{X_t}(X^*) \rangle \leq \mathrm{dist}^2(X_t, X^*) - \mathrm{dist}^2(X_{t+1}, X^*) + \eta^2 \|\mathrm{grad} f(X_t)\|^2.$$

Combining with (7.3.0.3), we get

$$\Delta_t \leq \frac{1}{4C_0\eta}(\mathrm{dist}^2(X_t, X^*) - \mathrm{dist}^2(X_{t+1}, X^*)) + \frac{\eta}{4C_0} \|\mathrm{grad} f(X_t)\|^2. \tag{7.3.0.6}$$

Now multiplying (7.3.0.5) by $\frac{1}{C_0}$ and summing with (7.3.0.6) gives

$$\frac{1}{C_0}\Delta_{t+1} - \left(\frac{1}{C_0} - 1\right)\Delta_t \leq \frac{1}{4C_0\eta}(\text{dist}^2(X_t, X^*) - \text{dist}^2(X_{t+1}, X^*)) + \frac{1}{C_0}\left(-\eta + \frac{L}{2}\eta^2 + \frac{\eta}{4}\right)\|\text{grad}f(X_t)\|^2. \quad (7.3.0.7)$$

By assumption, we have $\eta \leq C_0/L$, where $0 < C_0 = (1+\cos(\text{dist}(X_0, X^*)))/4 \leq \frac{1}{2}$ and $L > 0$. Since

$$\frac{\eta}{C_0}\left(-1 + \frac{L}{2}\eta + \frac{1}{4}\right) \leq \frac{\eta}{C_0}\left(\frac{C_0}{2} - \frac{3}{4}\right) \leq -\frac{1}{2}\frac{\eta}{C_0} < 0.$$

Inequality (7.3.0.7) can be simplified to

$$\frac{1}{C_0}\Delta_{t+1} - \left(\frac{1}{C_0} - 1\right)\Delta_t \leq \frac{1}{4C_0\eta}(\text{dist}^2(X_t, X^*) - \text{dist}^2(X_{t+1}, X^*)).$$

Summing from 0 to $t-1$ gives

$$\frac{1}{C_0}\Delta_t + \sum_{s=1}^{t-1}\Delta_s - \left(\frac{1}{C_0} - 1\right)\Delta_0 \leq \frac{1}{4C_0\eta}\left(\text{dist}^2(X_0, X^*) - \text{dist}^2(X_t, X^*)\right).$$

From Proposition 7.4 (and its implication presented in the first bullet point of Proposition 1.21 with $y \rightsquigarrow X_0$ and $x \rightsquigarrow X^*$), we get

$$\Delta_0 \leq \frac{L}{2}\text{dist}^2(X_0, X^*).$$

Combining these two inequalities leads to

$$\frac{1}{C_0}\Delta_t + \sum_{s=0}^{t-1}\Delta_s \leq \frac{1}{C_0}\Delta_0 + \frac{1}{4C_0\eta}\text{dist}^2(X_0, X^*) \leq \frac{1}{2C_0}\left(L + \frac{1}{2\eta}\right)\text{dist}^2(X_0, X^*).$$

Since (7.3.0.5) holds for all $t \geq 0$, it also implies $\Delta_t \leq \Delta_s$ for all $0 \leq s \leq t$. Substituting

$$t\Delta_t \leq \sum_{s=0}^{t-1}\Delta_s$$

into the inequality from above, we obtain

$$\Delta_t \leq \frac{1}{2C_0}\frac{L + \frac{1}{2\eta}}{\frac{1}{C_0} + t}\text{dist}^2(X_0, X^*) = \frac{L + \frac{1}{2\eta}}{2(C_0 t + 1)}\text{dist}^2(X_0, X^*).$$

After substituting C_0, the last inequality provides the desired convergence rate. ■

8 The importance of weak-quasi-strong convexity in optimization

As promised in the introduction, we show that WQSC is a necessary property for gradient descent applied to an L-smooth optimization problem to have linear convergence with respect to *distances of the iterates to some optimum*. A similar result but for the connection between the PL condition and linear convergence with respect to function values has been proved in [1] (Theorem 5). We follow here the exposition of our work [7] with minor modifications.

8.1 Introduction

As discussed in Section 1.2.2, a function is said to satisfy a WQSC condition if it satisfies Definition 1.6. Notice that Definition 1.6 assumes that the optimum is unique in the domain of interest. This definition (or rather Definition 1.22 about geodesic WQSC) is enough for the cases of the symmetric eigenvalue problem (Theorem 2.7) and polar decomposition (Proposition 7.3), as the optima in these cases are isolated (given that the spectral gap and the smallest singular value are positive respectively). We give here a slightly more general definition that includes also the case that the optima form a continuum. This type of definition is more popular in the literature, see for instance [78] (Definition 1) or [56] (Appendix A).

Definition 8.1 (Weak-quasi-strong convexity (WQSC)) *A function $f\colon \mathbb{R}^n \to \mathbb{R}$ with a convex set of global optima $X^* := \operatorname{argmin}_{x\in\mathbb{R}^n} f(x)$ is called (a,μ)-weak-quasi-strongly convex (WQSC) in a set $E \subseteq \mathbb{R}^n$, if there exist constants $a, \mu > 0$ such that*

$$f(x) - f^* \leq \frac{1}{a}\langle \nabla f(x), x - x_p\rangle - \frac{\mu}{2}\|x - x_p\|^2, \quad \forall x \in E,$$

where x_p is the projection of x onto X^*.

Remark 8.1 *Notice that the set of optima X^* is assumed to be convex. This assumption is necessary to ensure that the projection onto this set is well-defined. An interesting class of non-convex functions with convex set of optima is* quasi-convex *functions (all level sets of a quasi-convex function are convex, thus also the set of optima).*

In this section, we will be refering to (a,μ)-WQSC property in the slightly more general sense of Definition 8.1.

Proposition 8.2 *If f is (a,μ)-WQSC in a set E, then it also satisfies the PL condition*

$$\|\nabla f(x)\|^2 \geq 2\mu a^2 (f(x) - f^*)$$

in E.

Proof The proof is similar to the one of Lemma 3.2 in [25]. For completeness though we re-analyze it as we now allow multiple global optima.

If f is (a, μ)-WQSC, then we have

$$f(x) - f^* \leq \frac{1}{a}\langle \nabla f(x), x - x_p \rangle - \frac{\mu}{2}\|x - x_p\|^2, \quad \forall x \in E,$$

where x_p is the projection of x onto the set of global optima.

We can write

$$\langle \nabla f(x), x - x_p \rangle \leq \frac{\rho}{2}\|\nabla f(x)\|^2 + \frac{1}{2\rho}\|x - x_p\|^2,$$

for all $\rho > 0$.

Combining the two inequalities, we get

$$f(x) - f^* \leq \frac{\rho}{2a}\|\nabla f(x)\|^2 + \frac{1}{2a\rho}\|x - x_p\|^2 - \frac{\mu}{2}\|x - x_p\|^2.$$

Choosing $\rho = \frac{1}{a\mu}$, the two last terms in the right hand side cancel out, and the inequality becomes

$$f(x) - f^* \leq \frac{1}{2a^2\mu}\|\nabla f(x)\|^2, \quad \forall x \in E,$$

which gives the desired result after a rearrangement.

■

WQSC in the form of Definition 8.1 can guarantee linear convergence of the gradient descent algorithm with respect to the distances of the iterates to the set of of optima X^*. We recall an iterate of the gradient descent as

$$\tilde{x} = x - \eta \nabla f(x). \tag{8.1.0.1}$$

Proposition 8.3 *Consider the optimization problem*

$$\min_{x \in \mathbb{R}^n} f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is L-smooth and (a, μ)-WQSC in E. An iterate $\tilde{x}$ of (8.1.0.1) *starting from $x \in E$ with step size $0 \leq \eta \leq a/L$ satisfies*

$$\|\tilde{x} - \tilde{x}_p\|^2 \leq (1 - a\mu\eta)\|x - x_p\|^2.$$

Here, $\tilde{x}_p$ is the projection of $\tilde{x}$ onto $X^* = \operatorname{argmin}_{x \in \mathbb{R}^n} f(x)$, while x_p is that of x.

Proof The proof is a simple adaptation of Lemma 4.2 in [25]. The difference is that, in this result, the global optimum is not necessarily unique. We state it here for completeness.

We inspect the quantity $\|\tilde{x} - x_p\|^2$. We have

$$\|\tilde{x} - x_p\|^2 = \|x - \eta\nabla f(x) - x_p\|^2 = \|x - x_p\|^2 - 2\eta\langle\nabla f(x), x - x_p\rangle + \eta^2\|\nabla f(x)\|^2. \tag{8.1.0.2}$$

Notice that since f is L-smooth, we have (by Proposition 1.2) that

$$f(x) - f^* \geq \frac{1}{2L}\|\nabla f(x)\|^2,$$

By (a, μ)-WQSC of f (Definition 8.1), we have

$$-\frac{1}{a}\langle\nabla f(x), x - x_p\rangle \leq f^* - f(x) - \frac{\mu}{2}\|x - x_p\|^2$$

and combining with the previous inequality we get

$$-\frac{1}{a}\langle\nabla f(x), x - x_p\rangle \leq -\frac{1}{2L}\|\nabla f(x)\|^2 - \frac{\mu}{2}\|x - x_p\|^2.$$

We now multiply this inequality by $2\eta a$ on both sides:

$$-2\eta\langle\nabla f(x), x - x_p\rangle \leq -\frac{\eta a}{L}\|\nabla f(x)\|^2 - \eta\mu a\|x - x_p\|^2.$$

Substituting in (8.1.0.2), we get

$$\|\tilde{x} - x_p\|^2 \leq (1 - a\mu\eta)\|x - x_p\|^2 + \left(\eta^2 - \frac{\eta a}{L}\right)\|\nabla f(x)\|^2$$

and since $0 \leq \eta \leq \frac{a}{L}$, we have

$$\|\tilde{x} - x_p\|^2 \leq (1 - a\mu\eta)\|x - x_p\|^2.$$

By noticing that $\|\tilde{x} - \tilde{x}_p\| \leq \|\tilde{x} - x_p\|$ since $\tilde{x}_p$ is the projection of $\tilde{x}$ to the set of optima, we get the desired result. ■

8.2 Necessity of WQSC

We now pass to the main result of this section, which is essentially the inverse of Proposition 8.3. That is to say, WQSC is in some sense the bare minimum that an L-smooth optimization problem must satisfy, such that gradient descent converges linearly with respect to intrinsic distances. The backbone of the proof is the same as Theorem 5 in [7], but it has two small differences in order to make it work for the case of WQSC in the sense of Definition 8.1: i) the contraction quantity in the linear rate is assumed to be proportional to the step-size, ii) a limit argument is used at the end, examining the inequality derived in [7] for arbitrarily small step sizes. For this limit argument to work, we need to assume a linear convergence rate for all step sizes η close to 0. This assumption is (totally) realistic though as it is supported by the conclusion of Proposition 8.3.

Theorem 8.4 *Let $f\colon \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable, bounded below, L-smooth (see Definition 1.1) and the set of its global optima X^* is convex. Consider the optimization problem*

$$\min_{x\in\mathbb{R}^n} f(x).$$

Assume that there exist constants $\overline{\eta}$ and d such that the new iterate $\tilde{x}$ of (8.1.0.1) started from any $x \in E \subseteq \mathbb{R}^n$ and with any step size $\eta \in (0, \overline{\eta})$ satisfies

$$\|\tilde{x} - \tilde{x}_p\|^2 \leq (1 - d\eta)\|x - x_p\|^2,$$

where x_p and $\tilde{x}_p$ are the projections of x and $\tilde{x}$ respectively onto X^*. Then, f is (a, μ)-WQSC in E with parameters

$$a := \frac{d}{2L} \ , \ \mu := \frac{L}{2}.$$

Proof Let $x \in E$ and $\tilde{x}$ the result of one iteration of gradient descent (8.1.0.1).

We first rewrite the term $\|\tilde{x} - \tilde{x}_p\|^2$:

$$\begin{aligned}\|\tilde{x} - \tilde{x}_p\|^2 &= \|x - \eta\nabla f(x) - \tilde{x}_p\|^2 \\ &= \|x - \tilde{x}_p\|^2 - 2\eta\langle\nabla f(x), x - \tilde{x}_p\rangle + \eta^2\|\nabla f(x)\|^2.\end{aligned}$$

For ease of notation, we set $c := d\eta$. This equality together with the contraction assumption gives

$$\|x-\tilde{x}_p\|^2-2\eta\langle\nabla f(x), x-\tilde{x}_p\rangle+\eta^2\|\nabla f(x)\|^2 \leq (1-c)\|x-x_p\|^2 \leq (1-c)\|x-\tilde{x}_p\|^2.$$

The second inequality follows from the fact that x_p is the projection of x onto X^* (as defined in Definition 8.1). The derived inequality can thus be rewritten as

$$2\eta\langle\nabla f(x), x - \tilde{x}_p\rangle \geq c\|x - \tilde{x}_p\|^2 + \eta^2\|\nabla f(x)\|^2. \tag{8.2.0.1}$$

Next, we use the inequality

$$\langle y, z\rangle \leq \frac{\rho}{2}\|y\|^2 + \frac{1}{2\rho}\|z\|^2$$

that holds for all $y, z \in \mathbb{R}^n$ and any $\rho > 0$ to obtain

$$\frac{\rho}{2}\|\nabla f(x)\|^2 \geq \langle\nabla f(x), x - \tilde{x}_p\rangle - \frac{1}{2\rho}\|x - \tilde{x}_p\|^2.$$

Multiplying both sides by $\frac{2\eta^2}{\rho}$, we get

$$\eta^2\|\nabla f(x)\|^2 \geq \frac{2\eta^2}{\rho}\langle\nabla f(x), x - \tilde{x}_p\rangle - \frac{\eta^2}{\rho^2}\|x - \tilde{x}_p\|^2.$$

Combining this with (8.2.0.1), we get

$$2\eta\langle\nabla f(x), x-\tilde{x}_p\rangle \geq c\|x-\tilde{x}_p\|^2 + \frac{2\eta^2}{\rho}\langle\nabla f(x), x-\tilde{x}_p\rangle - \frac{\eta^2}{\rho^2}\|x-\tilde{x}_p\|^2,$$

or equivalently

$$\left(2\eta - \frac{2\eta^2}{\rho}\right)\langle\nabla f(x), x-\tilde{x}_p\rangle \geq \left(c - \frac{\eta^2}{\rho^2}\right)\|x-\tilde{x}_p\|^2.$$

Since the last inequality holds for any $\rho > 0$, we choose $\rho := \frac{2\eta}{\sqrt{c}}$ so that it becomes

$$2\eta\left(1-\frac{\sqrt{c}}{2}\right)\langle\nabla f(x), x-\tilde{x}_p\rangle \geq \frac{3c}{4}\|x-\tilde{x}_p\|^2 = \frac{c}{4}\|x-\tilde{x}_p\|^2 + \frac{c}{2}\|x-\tilde{x}_p\|^2.$$

By L-smoothness of f we have (substitute $y \rightsquigarrow x, x \rightsquigarrow \tilde{x}_p$ in Equation (2.1.6) of Theorem 2.1.5 of [80])

$$\|x-\tilde{x}_p\|^2 \geq \frac{2}{L}(f(x)-f^*),$$

and using that to bound the last term of the previous inequality, we have

$$2\eta\left(1-\frac{\sqrt{c}}{2}\right)\langle\nabla f(x), x-\tilde{x}_p\rangle \geq \frac{c}{4}\|x-\tilde{x}_p\|^2 + \frac{c}{L}(f(x)-f^*).$$

Rearranging, we get

$$f(x)-f^* \leq 2L\eta\frac{1-\frac{\sqrt{c}}{2}}{c}\langle\nabla f(x), x-\tilde{x}_p\rangle - \frac{L}{4}\|x-\tilde{x}_p\|^2.$$

Since $c = d\eta$, we substitute and obtain

$$f(x)-f^* \leq 2L\frac{1-\frac{\sqrt{d\eta}}{2}}{d}\langle\nabla f(x), x-\tilde{x}_p\rangle - \frac{L}{4}\|x-\tilde{x}_p\|^2,$$

for all $\eta \in (0, \bar{\eta})$.

Taking the limit $\eta \longrightarrow 0$ in both sides of this inequality, we have that $\tilde{x} \longrightarrow x$ and, since the metric projection onto a convex set is a continuous function, also that $\tilde{x}_p \longrightarrow x_p$. Putting everything together, we have the desired result:

$$f(x)-f^* \leq \frac{2L}{d}\langle\nabla f(x), x-x_p\rangle - \frac{L}{4}\|x-x_p\|^2.$$

■

We believe that Theorem 8.4 is a deep result, since it states that an L-smooth optimization problem is solvable via gradient descent with a linear convergence rate with respect to distances of the iterates to the set of optima if and only if it is weak-quasi-strongly convex. A valuable role is played by the scaling depending on the constant a. Versions of WQSC have mainly appeared in the literature without the $1/a$ scaling in front of the inner product in Definition 8.1 see ([78], Definition 1 and [56], Appendix A). This scaling gives a weaker property (what we call "quasi" in naming WQSC), which is still sufficient though to guarantee linear convergence of gradient descent with respect to distances of the iterates to the set of optima (Proposition 8.3). Moreover, it has enough expressive power to include important problems like the symmetric eigenvalue problem and polar decomposition. Even more importantly, it is also a necessary property for this type of convergence (the result of Theorem 8.4 would not be possible without the $1/a$ scaling).

Now we give a simple but fundamental corollary that connects the two types of convergence (with respect to distances and function values).

Corollary 8.5 *Consider a function $f : \mathbb{R}^n \to \mathbb{R}$ and the problem*

$$\min_{x \in \mathbb{R}^n} f(x).$$

If f is L-smooth with a convex set of global optima and an iterate $\tilde{x}$ of gradient descent (8.1.0.1) *starting from any $x \in \mathbb{R}^n$ with any step size $\eta \in (0, 2/L)$ satisfies*

$$\|\tilde{x} - \tilde{x}_p\|^2 \leq (1 - d\eta)\|x - x_p\|^2,$$

for a constant $d \in (0, 1/\eta)$, then an iterate $\bar{x}$ of (8.1.0.1) *starting from x with step size $\bar{\eta} \in (0, 2/L)$ satisfies*

$$f(\bar{x}) - f^* \leq \left(1 - \left(\frac{2}{L} - \bar{\eta}\right)\frac{d^2\bar{\eta}}{8}\right)(f(x) - f^*).$$

Proof Since an iterate of gradient descent contracts with respect to the distance from X^*, Theorem 8.4 implies that f is (a, μ)-WQSC, with $a = \frac{d}{2L}$ and $\mu = \frac{L}{2}$. By Proposition 8.2, we have that f satisfies the PL condition

$$\|\nabla f(x)\|^2 \geq 2\mu a^2(f(x) - f^*) = 2\frac{L}{2}\frac{d^2}{4L^2}(f(x) - f^*) = \frac{d^2}{4L}(f(x) - f^*).$$

By Proposition 1.5 for a general step size $\bar{\eta}$ (Proposition 1.5 is for $\bar{\eta} = \frac{1}{L}$ but the adaptation for a general step size is straightforward), we have that

$\bar{x} = x - \bar{\eta}\nabla f(x)$ satisfies

$$f(\bar{x}) - f^* \leq \left(1 - \bar{\eta}(2 - \bar{\eta}L)\frac{d^2}{8L}\right)(f(x) - f^*)$$
$$= \left(1 - \left(\frac{2}{L} - \bar{\eta}\right)\frac{d^2\bar{\eta}}{8}\right)(f(x) - f^*).$$

■

Remark: In Corollary 8.5 we pass from linear convergence with respect to distances to linear convergence with respect to function values. We lose something from the sharpness of the contraction though: if $\bar{\eta} = \frac{1}{L}$ and $\tilde{x} \equiv \bar{x} = x - \bar{\eta}\nabla f(x)$, then starting from a rate

$$\|\tilde{x} - \tilde{x}_p\|^2 \leq (1 - d\bar{\eta})\|x - x_p\|^2$$

yields to a rate

$$f(\bar{x}) - f^* \leq \left(1 - \frac{d^2\bar{\eta}^2}{8}\right)(f(x) - f^*).$$

As $d\bar{\eta} < 1$, the latter rate is slower.

8.3 The manifold case

In this section, we extend our main result to Riemannian gradient descent in a complete Riemannian manifold $\mathcal{M}$ of sectional curvatures bounded from above. This analysis mainly consists of combining the technique of Theorem 8.4 with standard geometric bounds, it is still a valuable result though since the main problems that concern us in this thesis (i.e. the symmetric eigenvalue problem and polar decomposition) are naturally posed on some manifold. For completeness, we start first with a general analysis of Riemannian gradient descent under geodesic L-smoothness and WQSC (Lemma 2.9 and Proposition 7.6 take into account that the relevant manifold is of nonnegative sectional curvatures). Geodesic L-smoothness has been defined in Definition 1.19, while geodesic WQSC in Definition 1.22. However, here we go with a slightly more general definition allowing multiple global optima forming a geodesically convex set in the spirit of Definition 8.1.

Definition 8.6
- *A geodesically convex subset X^* of a complete Riemannian manifold M is one such that every two points inside it are connected by some geodesic.*
- *A function $f : \mathcal{M} \to \mathbb{R}$ defined in a complete manifold $\mathcal{M}$ and with a geodesically convex set of global optima $X^* = \operatorname{argmin}_{x\in M} f(x)$ is called*

geodesically (a,μ)-WQSC in a subset $E \subseteq \mathcal{M}$, if the projection of any point of E onto X^* is unique and it holds

$$f(x) - f^* \leq \frac{1}{a}\langle \text{grad} f(x), -\text{Log}_x(x_p)\rangle - \frac{\mu}{2}\text{dist}^2(x, x_p), \quad \forall x \in E,$$

where x_p is the projection of x onto X^*.

Remark: Pay attention on the statement "if the projection of any point of E onto X^* is unique". In Riemannian manifolds, the uniqueness of projection becomes trickier. We adopt this simple statement in order to avoid getting too deep into the matter, but, in general, the uniqueness of projection of a point p to a set X^* is equivalent with the strict convexity of the distance function $d : X^* \to \mathbb{R}$:

$$d(x) = \text{dist}(x, p).$$

For manifolds of nonpositive sectional curvatures (i.e. Euclidean and hyperbolic spaces), the distance function from any point to any closed and geodesically convex set is always strictly convex, thus the projection is always unique. In manifolds that attain some positive sectional curvatures though, the situation is not that simple. Take for example a sphere, choose two poles, consider the northern hemisphere as the closed and geodesically convex subset of interest, and try to project the south pole onto it. Then, all the points in the equator are potential candidates. A general bound about the convexity of the distance function in manifolds of positive curvatures is given in [10] (Corollary 2.1).

Similarly as before, we recall an iterate of Riemannian gradient descent as

$$\tilde{x} = \exp_x(-\eta \text{grad} f(x)) \tag{8.3.0.1}$$

Proposition 8.7 *Let $\mathcal{M}$ be a complete Riemannian manifold of sectional curvatures bounded from below by $k_{\min}$. Consider the optimization problem*

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

where $f : \mathcal{M} \to \mathbb{R}$ is geodesically L-smooth in $\mathcal{M}$ and (a,μ)-WQSC in a subset $E \subseteq \mathcal{M}$ as in Definition 8.6. If $\tilde{x}$ is produced by one iterate of Riemannian gradient descent (8.3.0.1) starting from $x \in E$ with $\eta \leq \frac{a}{\zeta L}$, where ζ is defined as

$$\zeta := \begin{cases} \frac{\sqrt{-k_{\min}}\text{dist}(x,x_p)}{\tanh(\sqrt{-k_{\min}}\text{dist}(x,x_p))} & , k_{\min} < 0 \\ 1 & , k_{\min} \geq 0, \end{cases}$$

then we have

$$\text{dist}^2(\tilde{x}, \tilde{x}_p) \leq (1 - a\mu\eta)\text{dist}^2(x, x_p),$$

where x_p is the unique projection of x onto X^*, while $\tilde{x}_p$ is some projection of $\tilde{x}$ onto X^*.

Proof Take an arbitrary $x \in E$ and $\tilde{x}$ the result of one iterate of Riemannian gradient descent (8.3.0.1).

By Lemma 6 in [118] combined with Lemma 2 in [10] (applied to the geodesic triangle $\Delta x\tilde{x}x_p$), we have that

$$\text{dist}^2(\tilde{x}, x_p) \leq \zeta \text{dist}^2(x, \tilde{x}) + \text{dist}^2(x, x_p) - 2\langle \text{Log}_x(\tilde{x}), \text{Log}_x(x_p)\rangle, \quad (8.3.0.2)$$

where ζ is

$$\zeta := \begin{cases} \frac{\sqrt{-k_{\min}}\text{dist}(x,x_p)}{\tanh(\sqrt{-k_{\min}}\text{dist}(x,x_p))} & , k_{\min} < 0 \\ 1 & , k_{\min} \geq 0. \end{cases}$$

By noticing that $\text{Log}_x(\tilde{x}) = -\eta \text{grad} f(x)$ by the structure of Riemannian gradient descent, we can rewrite this inequality as

$$\text{dist}^2(\tilde{x}, x_p) \leq \text{dist}^2(x, x_p) - 2\eta\langle \text{grad} f(x), -\text{Log}_x(x_p)\rangle + \zeta\eta^2 \|\text{grad} f(x)\|^2. \quad (8.3.0.3)$$

By geodesic (a, μ)-WQSC (Definition 8.6), we have that

$$-2\eta\langle \text{grad} f(x), -\text{Log}_x(x_p)\rangle \leq -2\eta a(f(x) - f^*) - a\mu\eta \text{dist}^2(x, x_p).$$

Applying L-smoothness, we have

$$-2\eta\langle \text{grad} f(x), -\text{Log}_x(x_p)\rangle \leq -\frac{\eta a}{L}\|\text{grad} f(x)\|^2 - a\mu\eta \text{dist}^2(x, x_p).$$

Plugging that in (8.3.0.3), we obtain

$$\text{dist}^2(\tilde{x}, x_p) \leq (1 - a\mu\eta)\text{dist}^2(x, x_p) + \left(\zeta\eta^2 - \frac{\eta a}{L}\right)\|\text{grad} f(x)\|^2.$$

Since $\eta \leq \frac{a}{\zeta L}$, we have $\zeta\eta^2 - \frac{\eta a}{L} \leq 0$, thus

$$\text{dist}^2(\tilde{x}, x_p) \leq (1 - a\mu\eta)\text{dist}^2(x, x_p).$$

By definition of $\tilde{x}_p$ and x_p, we have

$$\text{dist}(\tilde{x}, \tilde{x}_p) \leq \text{dist}(\tilde{x}, x_p)$$

and the desired result follows. ■

Remark 8.2 *As it is evident by previous works in the field [10, 118],* convergence is harder in the case of lower curvatures *(ζ is* 1 *if curvatures are nonnegative but larger than* 1 *if curvatures are negative).*

Before passing to the Riemannian extension of Theorem 8.4, we need an auxiliary geometric result similar to Lemma 6 in [118], which can be found in Corollary 2.1 of [10].

Lemma 8.8 *Let Δabc be a geodesic triangle (i.e. a triangle whose sides are geodesics) in a complete manifold of sectional curvatures bounded from above by $k_{\max}$. If $k_{\max} > 0$, we assume in addition that the lengths of the sides of this triangle are less than $\pi/\sqrt{k_{\max}}$. Then*

$$\operatorname{dist}^2(a,c) \geq \delta \cdot \operatorname{dist}^2(b,c) + 2\langle \operatorname{Log}_b(a), \operatorname{Log}_b(c)\rangle + \operatorname{dist}^2(a,b).$$

where

$$\delta = \begin{cases} \frac{\sqrt{k_{\max}}\operatorname{dist}(a,q)}{\tan(\sqrt{k_{\max}}\operatorname{dist}(a,q))} & , k_{\max} > 0 \\ 1 & , k_{\max} \leq 0, \end{cases}$$

with q being some point on the geodesic bc.

We use this lemma to prove a Riemannian analogue of Theorem 8.4:

Theorem 8.9 *Consider the L-smooth optimization problem*

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

with $\mathcal{M}$ being a complete Riemannian manifold of sectional curvatures bounded from above by $k_{\max}$. Also assume that the set of optima $X^* = \operatorname{argmin}_{x\in M} f(x)$ is geodesically convex.
Assume that a step of Riemannian gradient descent (8.3.0.1) *starting from any point $x \in E \subseteq M$ satisfies*

$$\operatorname{dist}^2(\tilde{x}, \tilde{x}_p) \leq (1 - d\eta)\operatorname{dist}^2(x, x_p)$$

for some constant $d > 0$ and any $\eta \in (0, \bar{\eta}), \bar{\eta} > 0$. If $k_{\max} > 0$, we assume also that

$$E \subseteq \left\{ x \in M \middle| \operatorname{dist}(x, x_p) < \frac{\pi}{2\sqrt{k_{\max}}} \right\} .$$

Then f is geodesically (a, μ)-WQSC in E, with

$$a := \frac{d}{2L},\ \mu := \frac{L}{2}.$$

Proof We fix an arbitrary point $x \in E$ and consider $\tilde{x}$ to be the result of one iterate of Riemannian gradient descent (8.3.0.1).

We first bound $\operatorname{dist}^2(\tilde{x}, \tilde{x}_p)$ using Lemma 8.8 in the geodesic triangle $\Delta x\tilde{x}\tilde{x}_p$:

$$\operatorname{dist}^2(\tilde{x}, \tilde{x}_p) \geq \delta \cdot \operatorname{dist}^2(x, \tilde{x}) + \operatorname{dist}^2(x, \tilde{x}_p) - 2\langle \operatorname{Log}_x(\tilde{x}), \operatorname{Log}_x(\tilde{x}_p)\rangle,$$

where

$$\delta = \begin{cases} \sqrt{k_{\max}}\operatorname{dist}(q, \tilde{x}_p)\cot(\sqrt{k_{\max}}\operatorname{dist}(q, \tilde{x}_p)) & , k_{\max} > 0 \\ 1 & , k_{\max} \leq 0, \end{cases}$$

with q being some point in the geodesic connecting x and $\tilde{x}$.

This inequality together with the assumed contraction (and after setting $c := d\eta$) gives

$$\delta \cdot \mathrm{dist}^2(x, \tilde{x}) + \mathrm{dist}^2(x, \tilde{x}_p) - 2\langle \mathrm{Log}_x(\tilde{x}), \mathrm{Log}_x(\tilde{x}_p)\rangle \leq (1-c)\mathrm{dist}^2(x, x_p) \tag{8.3.0.4}$$

$$\leq (1-c)\mathrm{dist}(x, \tilde{x}_p). \tag{8.3.0.5}$$

Even when $k_{\max} > 0$, δ can be lower bounded as follows:
the function $x \to x\cot(x)$ is decreasing if $x > 0$, thus it suffices to bound $\mathrm{dist}(q, \tilde{x}_p)$ from above. To that end, we have

$$\mathrm{dist}(q, \tilde{x}_p) \leq \mathrm{dist}(\tilde{x}, \tilde{x}_p) + \mathrm{dist}(\tilde{x}, q) \leq \mathrm{dist}(x, x_p) + \mathrm{dist}(x, \tilde{x}).$$

Since $\tilde{x} = \mathrm{Exp}_x(-\eta \mathrm{grad} f(x))$, we have that $\mathrm{Log}_x(\tilde{x}) = -\eta \mathrm{grad} f(x)$ and $\mathrm{dist}(x, \tilde{x}) = \eta \|\mathrm{grad} f(x)\|$. Moreover, by L-smoothness of f, we have that $\|\mathrm{grad} f(x)\| \leq L\mathrm{dist}(x, x_p)$.

Using all these facts, we can bound $\mathrm{dist}(q, x_p)$ as

$$\mathrm{dist}(q, \tilde{x}_p) \leq (1 + \eta L)\, \mathrm{dist}(x, x_p).$$

This implies

$$\delta \geq \bar{\delta}(\eta) = \begin{cases} \frac{(1+\eta L)\sqrt{k_{\max}}\mathrm{dist}(x,x_p)}{\tan((1+\eta L)\sqrt{k_{\max}}\mathrm{dist}(x,x_p))} & , k_{\max} > 0 \\ 1 & , k_{\max} \leq 0, \end{cases}$$

for η sufficiently small.

This bound can be potentially negative, but in the limit case that η becomes arbitrarily small, it becomes positive. This is because, if $k_{\max} > 0$, we have assumed that $\mathrm{dist}(x, x_p) < \frac{\pi}{2\sqrt{k_{\max}}}$. Thus, we fix an $\eta_0 > 0$, such that $\bar{\delta}(\eta) > 0$, for any $\eta \in (0, \eta_0)$. Since our assumed convergence rate holds for all step sizes η arbitrarily close to 0, we can continue the derivation assuming that $\eta < \eta_0$.

Using again that $\mathrm{Log}_x(\tilde{x}) = -\eta \mathrm{grad} f(x)$, we can rewrite inequality (8.3.0.4) as

$$2\eta\langle \mathrm{grad} f(x), -\mathrm{Log}_x(\tilde{x}_p)\rangle \geq c \cdot \mathrm{dist}^2(x, \tilde{x}_p) + \bar{\delta}(\eta)\eta^2 \|\mathrm{grad} f(x)\|^2. \tag{8.3.0.6}$$

Next, we use the inequality

$$\langle \alpha, \beta\rangle \leq \frac{\rho}{2}\|\alpha\|^2 + \frac{1}{2\rho}\|\beta\|^2$$

for any $\alpha, \beta \in T_x M$ and $\rho > 0$ and we obtain

$$\frac{\rho}{2}\|\mathrm{grad} f(x)\|^2 \geq \langle \mathrm{grad} f(x), -\mathrm{Log}_x(\tilde{x}_p)\rangle - \frac{1}{2\rho}\mathrm{dist}^2(x, \tilde{x}_p).$$

Multiplying both sides by $\frac{2\bar{\delta}(\eta)\eta^2}{\rho}$, we get

$$\bar{\delta}(\eta)\eta^2\|\text{grad}f(x)\|^2 \geq \frac{2\bar{\delta}(\eta)\eta^2}{\rho}\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle - \frac{\bar{\delta}(\eta)\eta^2}{\rho^2}\text{dist}^2(x,\tilde{x}_p).$$

Using equation (8.3.0.6), we get

$$2\eta\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle \geq$$
$$c\cdot\text{dist}^2(x,\tilde{x}_p) + \frac{2\bar{\delta}(\eta)\eta^2}{\rho}\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle - \frac{\bar{\delta}(\eta)\eta^2}{\rho^2}\text{dist}^2(x,\tilde{x}_p),$$

or equivalently

$$\left(2\eta - \frac{2\bar{\delta}(\eta)\eta^2}{\rho}\right)\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle \geq \left(c - \frac{\bar{\delta}(\eta)\eta^2}{\rho^2}\right)\text{dist}^2(x,\tilde{x}_p).$$

Since the last inequality holds for any $\rho > 0$, we can choose $\rho = 2\frac{\sqrt{\bar{\delta}(\eta)}\eta}{\sqrt{c}}$. Then it becomes

$$2\eta\left(1 - \frac{\sqrt{\bar{\delta}(\eta)}\sqrt{c}}{2}\right)\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle \geq$$
$$\frac{3c}{4}\text{dist}^2(x,\tilde{x}_p) = \frac{c}{4}\text{dist}^2(x,\tilde{x}_p) + \frac{c}{2}\text{dist}^2(x,\tilde{x}_p).$$

By geodesic L-smoothness, we have

$$\text{dist}^2(x,\tilde{x}_p) \geq \frac{2}{L}(f(x) - f^*),$$

and using that to bound the last term of the previous inequality, we have

$$2\eta\left(1 - \frac{\sqrt{\bar{\delta}(\eta)}\sqrt{c}}{2}\right)\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle \geq \frac{c}{4}\text{dist}^2(x,\tilde{x}_p) + \frac{c}{L}(f(x) - f^*).$$

Rearranging, we get

$$f(x) - f^* \leq 2L\eta\frac{1 - \frac{\sqrt{\bar{\delta}(\eta)}\sqrt{c}}{2}}{c}\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle - \frac{L}{4}\text{dist}^2(x,\tilde{x}_p).$$

Substituting $c = d\eta$, we get

$$f(x) - f^* \leq 2L\eta\frac{1 - \frac{\sqrt{\bar{\delta}(\eta)}\sqrt{d\eta}}{2}}{d\eta}\langle \text{grad}f(x), -\text{Log}_x(\tilde{x}_p)\rangle - \frac{L}{4}\text{dist}^2(x,\tilde{x}_p).$$

Taking the limit when $\eta \longrightarrow 0$, we get $\tilde{x}_p \longrightarrow x_p$, thus

$$f(x) - f^* \leq \frac{2L}{d}\langle \text{grad}f(x), -\text{Log}_x(x_p)\rangle - \frac{L}{4}\text{dist}^2(x,x_p).$$

This is the desired result.

■

9 Conclusion

9.1 Reflection on our contributions

We are confident that this thesis contributes meaningfully to the theory of both non-convex optimization and numerical linear algebra.

The reader interested primarily in optimization will perhaps view its main contribution as the thorough analysis of the weak-quasi-strong convexity property. This property was proved to be necessary and sufficient for linear convergence of gradient descent with respect to distances of the iterates to the set of optima, as the more well-known PL inequality, which has a similar behavior but with respect to function values. We also identified two important problems from linear algebra that serve as good examples of this structure.

The reader primarily interested in linear algebra will probably prioritize the justification of the tractability of the symmetric eigenvalue and polar decomposition problems through the aforementioned convexity-like structure, but also the practical contributions of this thesis. Highlights include a novel state-of-the-art theory for preconditioned eigenvalue solvers (Section 4) and the development of really competitive eigenvalue solvers (Section 5) that could be considered by all kinds of practitioners from now on.

The reader mostly interested in *Riemannian* optimization, i.e. the field of optimization over non-linear surfaces, will perhaps see this thesis as a success story for the very case of this field: non-convex problems in the Euclidean sense can be convex (or quasi-convex) in an intrinsic Riemannian sense, if they are posed properly over some Riemannian manifold.

9.2 Directions for future work

Fruitful directions for future work can easily be deduced directly from the topics treated in this thesis.

Section 4, for instance, deals only with the case of the basic preconditioned eigenvalue solver (PINVIT) and not with the state-of-the-art one (LOBPCG). It would be interesting to see whether a modification of this analysis can be applied also to LOBPCG. This is not clear to us at this point.

A main narrative in this thesis is the value of the derived convexity-like structures in analyzing eigenvalue or polar decomposition problems in noisy regimes. The main examples we gave are i) distributed scenaria with limited communication (Section 3) and ii) preconditioned eigenvalue solvers (Section 4), which are essentially perturbed versions of inverse iteration. Other important noisy regimes that are worth examining could be computing eigenvalues or polar factors via stochastic algorithms, or even "robust" re-formulations (see equation (7.1.0.1)).

Deviating a bit from the exact topics of this thesis, but staying inside the general philosophy, one could try to show some convexity-like structure for

the problem of optimally approximating a matrix with another matrix of fixed rank. This problem admits a closed-form solution via the truncated SVD of the matrix, which is essentially the projection onto the manifold of fixed rank matrices. Similarly, orthogonal Procrustes (Section 7) is the projection of a matrix onto the orthogonal group. It would not be too surprising if the problem of low rank approximation admits a similar structure, we expect the situation to be more involved though, as the manifold of fixed rank matrices has a much more complicated Riemannian structure compared to the orthogonal group. Another interesting problem over the Stiefel manifold is computing the polar factor of a *rectangular* matrix.

Taking the discussion of the previous paragraph a step further, while linear algebra offers an ecosystem of problems that are really interesting and important, deep learning has dominated the field of non-convex optimization the last few years. An important open problem in the theory of deep learning has to do with explaining its success: while all real-world deep neural networks are highly non-convex, training them using stochastic first-order methods has been proven surprisingly effective. We conjecture that this is the case due to various convexity-like structures that appear in these optimization problems. Results of this nature have long been appeared for the case of *over-parametrized* models [105]. Over-parametrization though is not always a realistic assumption. More recently, research started going beyond it [51]. Given the paramount importance of deep learning models in our society and economy, we believe that understanding their structure must be set priority by the non-convex optimization community. We personally wish to contribute in this direction in the years to come.

References

[1] Hadi Abbaszadehpeivasti, Etienne de Klerk, and Moslem Zamani. Conditions for linear convergence of the gradient method for non-convex optimization. *Optimization Letters*, 17(5):1105–1125, 2023. https://doi.org/10.1007/s11590-023-01981-2.

[2] P.-A. Absil, Robert Mahony, and Rodolphe Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. *Acta Applicandae Mathematicae*, 80(2):199–220, 2004. https://doi.org/10.1023/b:acap.0000013855.14971.91.

[3] P-A Absil, Robert Mahony, and Rodolphe Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, 2009.

[4] Kwangjun Ahn and Suvrit Sra. From nesterov's estimate sequence to riemannian acceleration. In *Conference on Learning Theory*, pages 84–118. PMLR, 2020.

[5] Kwangjun Ahn and Felipe Suarez. Riemannian perspective on matrix factorization. *arXiv:2102.00937*, 2021.

[6] Devrim Akca. Generalized procrustes analysis and its applications in photogrammetry. Technical report, ETH Zurich, 2003. https://doi.org/10.3929/ethz-a-004656648.

[7] Foivos Alimisis. Characterization of optimization problems that are solvable iteratively with linear convergence. *MTNS*, 2024. https://doi.org/10.1016/j.ifacol.2024.10.182.

[8] Foivos Alimisis, Peter Davies, Bart Vandereycken, and Dan Alistarh. Distributed principal component analysis with limited communication. *Advances in Neural Information Processing Systems*, 34, 2021.

[9] Foivos Alimisis, Daniel Kressner, Nian Shao, and Bart Vandereycken. A preconditioned inverse iteration with an improved convergence guarantee. *arXiv preprint arXiv:2412.14665*, 2024.

[10] Foivos Alimisis, Antonio Orvieto, Gary Bécigneul, and Aurelien Lucchi. A continuous-time perspective for modeling acceleration in riemannian optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1297–1307. PMLR, 2020.

[11] Foivos Alimisis, Antonio Orvieto, Gary Becigneul, and Aurelien Lucchi. Momentum improves optimization on Riemannian manifolds. In *International Conference on Artificial Intelligence and Statistics*, pages 1351–1359. PMLR, 2021.

[12] Foivos Alimisis, Yousef Saad, and Bart Vandereycken. Gradient-type subspace iteration methods for the symmetric eigenvalue problem. *SIAM Journal on Matrix Analysis and Applications*, 45(4):2360–2386, 2024. https://doi.org/10.1137/23M1590792.

[13] Foivos Alimisis and Bart Vandereycken. A convexity-like structure for polar decomposition with an application to distributed computing. *arXiv preprint arXiv:2412.13990*, 2024.

[14] Foivos Alimisis and Bart Vandereycken. Geodesic convexity of the symmetric eigenvalue problem and convergence of steepest descent. *Journal of Optimization Theory and Applications*, pages 1–40, 2024. https://doi.org/10.1007/s10957-024-02538-8.

[15] Foivos Alimisis, Simon Vary, and Bart Vandereycken. A Nesterov-style accelerated gradient descent algorithm for the symmetric eigenvalue problem. *arXiv preprint: 2406.18433*, 2024. https://doi.org/10.48550/arXiv.2406.18433.

[16] Foivos Alimisis, Simon Vary, and Bart Vandereycken. A nesterov-style accelerated gradient descent algorithm for the symmetric eigenvalue problem. *arXiv preprint arXiv:2406.18433*, 2024.

[17] Christopher G Baker. *Riemannian manifold trust-region methods with applications to eigenproblems*. The Florida State University, 2008.

[18] Thomas Bendokat, Ralf Zimmermann, and P-A Absil. A Grassmann manifold handbook: Basic geometry and computational aspects. *Advances in Computational Mathematics*, 50(1):6, 2024. https://doi.org/10.1007/s10444-023-10090-8.

[19] Christopher Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[20] Nicolas Boumal. Optimality conditions on the orthogonal group. https://www.racetothebottom.xyz/posts/optimality-orthogonal/.

[21] Nicolas Boumal. *An Introduction to Optimization on Smooth Manifolds*. Cambridge University Press, 2023. https://doi.org/10.1017/9781009166164.

[22] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. https://doi.org/10.1017/CBO9780511804441.

[23] Susanne C. Brenner and L. Ridgway Scott. *The mathematical theory of finite element methods*, volume 15 of *Texts in Applied Mathematics*. Springer, New York, third edition, 2008. https://doi.org/10.1007/978-0-387-75934-0.

[24] William L. Briggs, Van Emden Henson, and Steve F. McCormick. *A multigrid tutorial*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2000. https://doi.org/10.1137/1.9780898719505.

[25] Jingjing Bu and Mehran Mesbahi. A note on nesterov's accelerated method in nonconvex optimization: a weak estimate sequence approach. *arXiv preprint arXiv:2006.08548*, 2020.

[26] Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015. https://doi.org/10.1561/2200000050.

[27] A. Bunse-Gerstner, R. Byers, V. Mehrmann, and N. K. Nichols. Numerical computation of an analytic singular value decomposition of a matrix valued function. *Numerische Mathematik*, 60(1):1–39, 1991. https://doi.org/10.1007/BF01385712.

[28] Ralph Byers and Hongguo Xu. A new scaling for newton's iteration for the polar decomposition and its backward stability. *SIAM Journal on Matrix Analysis and Applications*, 30(2):822–843, 2008. https://doi.org/10.1137/070699895.

[29] Jeff Cheeger and David Ebin. *Comparison theorems in Riemannian geometry*, volume 9. North-Holland publishing company Amsterdam, 1975.

[30] Christopher Criscitiello and Nicolas Boumal. Negative curvature obstructs acceleration for strongly geodesically convex optimization, even with exact first-order oracles. In *Conference on Learning Theory*, pages 496–542. PMLR, 2022.

[31] Peter Davies, Vijaykrishna Gurunathan, Niusha Moshrefi, Saleh Ashkboos, and Dan Alistarh. New bounds for distributed mean estimation and variance reduction. In *International Conference on Learning Representations*, 2021.

[32] Timothy Davis and Yifan Hu. The University of Florida Sparse Matrix Collection. *ACM Trans. Math. Softw.*, 38(1), dec 2011. https://doi.org/10.1145/2049662.2049663.

[33] Dmitriy Drusvyatskiy and Adrian S. Lewis. Error bounds, quadratic growth, and linear convergence of proximal methods. *Mathematics of Operations Research*, 43(3):919–948, 2018. https://doi.org/10.1287/moor.2017.0889.

[34] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[35] Alan Edelman, Tomás A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1999. https://doi.org/10.1137/S0895479895290954.

[36] Shira Faigenbaum-Golovin and Ingrid Daubechies. Studying morphological variation: Exploring the shape space in evolutionary anthropology. *arXiv preprint arXiv:2410.20040*, 2024.

[37] Walter Gander. On halley's iteration method. *The American Mathematical Monthly*, 92(2):131–134, 1985. https://doi.org/10.2307/2322644.

[38] Dan Garber, Ohad Shamir, and Nathan Srebro. Communication-efficient algorithms for distributed stochastic principal component analysis. In *International Conference on Machine Learning*, pages 1203–1212. PMLR, 2017.

[39] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.

[40] Sergey Guminov, Alexander Gasnikov, and Ilya Kuruzov. Accelerated methods for α-weakly-quasi-convex problems. *arXiv preprint arXiv:1710.00797*, 2017.

[41] Linus Hamilton and Ankur Moitra. A no-go theorem for robust acceleration in the hyperbolic plane. *Advances in Neural Information Processing Systems*, 34:3914–3924, 2021.

[42] Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. *Advances in neural information processing systems*, 27, 2014.

[43] Magnus Hestenes and William Karush. A method of gradients for the calculation of the characteristic roots and vectors of a real symmetric matrix. *Journal of Research of the National Bureau of Standards*, 1951. https://doi.org/10.6028/jres.047.008.

[44] Nicholas Higham. Computing the polar decomposition—with applications. *SIAM Journal on Scientific and Statistical Computing*, 7(4):1160–1174, 1986. https://doi.org/10.1137/0907079.

[45] Nicholas Higham. Functions of matrices: Theory and computation, 2008. https://doi.org/10.1137/1.9780898717778.

[46] Nicholas Higham and Sheung Cheng. Modifying the inertia of matrices arising in optimization. *Lin. Alg. Appl.*, 275–276:261–279, 1998. https://doi.org/10.1016/s0024-3795(97)10015-5.

[47] Roger Horn and Charles Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge ; New York, 2nd edition, 2012. https://doi.org/10.1017/CBO9781139020411.

[48] Roger Horn and Charles R. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, 1991.

[49] Samuel Horváth, Dmitry Kovalev, Konstantin Mishchenko, Sebastian Stich, and Peter Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *Optimization Methods and Software*, 38(1):91–106, 2023. https://doi.org/10.1080/10556788.2022.2117355.

[50] Long-Kai Huang and Sinno Pan. Communication-efficient distributed PCA by Riemannian optimization. In *International Conference on Machine Learning*, pages 4465–4474. PMLR, 2020.

[51] Rustem Islamov, Niccolò Ajroldi, Antonio Orvieto, and Aurelien Lucchi. Loss landscape characterization of neural networks without over-parametrization. *Advances in Neural Information Processing Systems*, 2024.

[52] Rustem Islamov, Xun Qian, and Peter Richtárik. Distributed second order methods with fast rates and compressed communication. In *International conference on machine learning*, pages 4617–4628. PMLR, 2021.

[53] Martin Jaggi, Virginia Smith, Martin Takác, Jonathan Terhorst, Sanjay Krishnan, Thomas Hofmann, and Michael I Jordan. Communication-efficient distributed dual coordinate ascent. *Advances in neural information processing systems*, 27, 2014.

[54] Michael Jordan, Tianyi Lin, and Emmanouil-Vasileios Vlatakis-Gkaragkounis. First-order algorithms for min-max optimization in geodesic metric spaces. *Advances in Neural Information Processing Systems*, 35:6557–6574, 2022.

[55] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976. https://doi.org/10.1107/S0567739476001873.

[56] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part I 16*, pages 795–811. Springer, 2016. https://doi.org/10.1007/978-3-319-46128-1_50.

[57] Charles Kenney and Alan J. Laub. On scaling newton's method for polar decomposition and the matrix sign function. In *1990 American Control Conference*, pages 2560–2564, 1990. https://doi.org/10.23919/ACC.1990.4791187.

[58] Jungbin Kim and Insoon Yang. Accelerated gradient methods for geodesically convex optimization: Tractable algorithms and convergence analysis. In *International Conference on Machine Learning*, pages 11255–11282. PMLR, 2022.

[59] Andrew V. Knyazev. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method. *SIAM journal on scientific computing*, 23(2):517–541, 2001. https://doi.org/10.1137/S1064827500366124.

[60] Andrew V. Knyazev, Merico Argentati, Ilya Lashuk, and Ovtchinnikov Evgueni. Block locally optimal preconditioned eigenvalue xolvers (blopex) in hypre and petsc. *SIAM Journal on Scientific Computing*, 29(5):2224–2239, 2007. https://doi.org/10.1137/060661624.

[61] Andrew V. Knyazev and Klaus Neymeyr. A geometric theory for preconditioned inverse iteration iii: A short and sharp convergence estimate for generalized eigenvalue problems. *Linear Algebra and its Applications*, 358(1-3):95–114, 2003. https://doi.org/10.1016/S0024-3795(01)00461-X.

[62] Andrew V. Knyazev and Alexander Shorokhodov. On exact estimates of the convergence rate of the steepest ascent method in the symmetric eigenvalue problem. *Linear Algebra and its Applications*, 154-156:245–257, 1991. https://doi.org/10.1016/0024-3795(91)90379-B.

[63] Janne H. Korhonen and Dan Alistarh. Towards tight communication lower bounds for distributed optimisation. *Advances in Neural Information Processing Systems*, 34:7254–7266, 2021.

[64] Daniel Kressner, Yuxin Ma, and Meiyue Shao. A mixed precision LOBPCG algorithm. *Numer. Algorithms*, 94(4):1653–1671, 2023. https://doi.org/10.1007/s11075-023-01550-9.

[65] Jacek Kuczynski and Henryk Wozniakowski. Estimating the largest eigenvalue by the power and Lanczos algorithms with a random start. *SIAM Journal on Matrix Analysis and Applications*, 1992. https://doi.org/10.1137/0613066.

[66] Rasmus Kyng and Sushant Sachdeva. Approximate Gaussian elimination for Laplacians—fast, sparse, and simple. In *57th Annual IEEE Symposium on Foundations of Computer Science—FOCS 2016*, pages 573–582. IEEE Computer Soc., Los Alamitos, CA, 2016.

[67] John M. Lee. *Introduction to Riemannian Manifolds*. Graduate Texts in Mathematics. Springer International Publishing, 2019.

[68] Claude Lemaréchal. Cauchy and the gradient method. 2012. https://doi.org/10.4171/DMS/6/27.

[69] Chi-Kwong Li and Roy Mathias. Inequalities on the singular values of an off-diagonal block of a Hermitian matrix. *Journal of Inequalities and Applications*, 3(2):137–142, 1999. https://doi.org/10.1155/S1025583499000090.

[70] Shengqiao Li. Concise Formulas for the Area and Volume of a Hyperspherical Cap. *Asian Journal of Mathematics & Statistics*, 4(1):66–70, 2011. https://doi.org/10.3923/ajms.2011.66.70.

[71] Shuang Li, Gongguo Tang, and Michael B. Wakin. Landscape correspondence of empirical and population risks in the eigendecomposition problem. *IEEE Transactions on Signal Processing*, 70:2985–2999, 2022. https://doi.org/10.1109/tsp.2022.3181333.

[72] Ross A. Lippert. Fixing two eigenvalues by a minimal perturbation. *Linear Algebra and its Applications*, 406:177–200, September 2005. https://doi.org/10.1016/j.laa.2005.04.004.

[73] David G. Luenberger and Yinyu Ye. *Linear and Nonlinear Programming*. Springer, New York, NY, 3rd edition edition, July 2008. https://doi.org/10.1007/978-0-387-74503-9.

[74] Sindri Magnússon, Hossein Shokri-Ghadikolaei, and Na Li. On maintaining linear convergence of distributed learning and optimization under limited communication. *IEEE Transactions on Signal Processing*, 68:6101–6116, 2020. https://doi.org/10.1109/IEEECONF44664.2019.9049052.

[75] David Martínez-Rubio and Sebastian Pokutta. Accelerated riemannian optimization: Handling constraints with a prox to bound geometric penalties. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 359–393. PMLR, 2023.

[76] Yuji Nakatsukasa, Zhaojun Bai, and François Gygi. Optimizing halley's iteration for computing the matrix polar decomposition. *SIAM Journal on Matrix Analysis and Applications*, 31(5):2700–2720, 2010. https://doi.org/10.1137/090774999.

[77] Artem Napov and Yvan Notay. An algebraic multigrid method with guaranteed convergence rate. *SIAM J. Sci. Comput.*, 34(2):A1079–A1109, 2012. https://doi.org/10.1137/100818509.

[78] Ion Necoara, Yurii Nesterov, and Francois Glineur. Linear convergence of first order methods for non-strongly convex optimization. *Mathematical Programming*, 175:69–107, 2019. https://doi.org/10.1007/s10107-018-1232-1.

[79] Yurii Nesterov. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983.

[80] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013. https://doi.org/10.1007/978-1-4419-8853-9.

[81] Yurii Nesterov, Alexander Gasnikov, Sergey Guminov, and Pavel Dvurechensky. Primal–dual accelerated gradient methods with small-dimensional relaxation oracle. *Optimization Methods and Software*, pages 1–38, 2020. https://doi.org/10.1080/10556788.2020.1731747.

[82] Klaus Neymeyr. A geometric theory for preconditioned inverse iteration. I. Extrema of the Rayleigh quotient. *Linear Algebra Appl.*, 322(1-3):61–85, 2001. https://doi.org/10.1016/S0024-3795(00)00239-1.

[83] Klaus Neymeyr. A geometric theory for preconditioned inverse iteration. II. Convergence estimates. *Linear Algebra Appl.*, 322(1-3):87–104, 2001. https://doi.org/10.1016/S0024-3795(00)00236-6.

[84] Klaus Neymeyr. A posteriori error estimation for elliptic eigenproblems. *Numer. Linear Algebra Appl.*, 9(4):263–279, 2002. https://doi.org/10.1002/nla.272.

[85] Klaus Neymeyr, Evgueni Ovtchinnikov, and Ming Zhou. Convergence analysis of gradient iterations for the symmetric eigenvalue problem. *SIAM J. Matrix Analysis Applications*, 32:443–456, 04 2011. https://doi.org/10.1137/100784928.

[86] Klaus Neymeyr and Ming Zhou. Iterative minimization of the Rayleigh quotient by block steepest descent iterations. *Numerical Linear Algebra with Applications*, 21(5):604–617, 2014. https://doi.org/10.1002/nla.1915.

[87] Jorge Nocedal and Stephen J Wright. *Numerical optimization*. Springer, 2006. https://doi.org/10.1007/978-0-387-40065-5.

[88] Yvan Notay. AGMG software and documentation. http://agmg.eu.

[89] Yvan Notay. An aggregation-based algebraic multigrid method. *Electron. Trans. Numer. Anal.*, 37:123–146, 2010.

[90] Erkki Oja and Juha Karhunen. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of mathematical analysis and applications*, 106(1):69–84, 1985. https://doi.org/10.1016/0022-247X(85)90131-3.

[91] Dianne P O'Leary, GW Stewart, and James S Vandergraft. Estimating the largest eigenvalue of a positive definite matrix. *Mathematics of Computation*, 33(148):1289–1292, 1979. https://doi.org/10.2307/2006463.

[92] Christopher C. Paige and Michael A. Saunders. Towards a generalized singular value decomposition. *SIAM Journal on Numerical Analysis*, 18(3):398–405, June 1981. https://doi.org/10.1137/0718026.

[93] Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal vychislitel'noi matematiki i matematicheskoi fiziki*, 3(4):643–653, 1963.

[94] Li Qiu, Yanxia Zhang, and Chi-Kwong Li. Unitarily invariant metrics on the Grassmann space. *SIAM Journal on Matrix Analysis and Applications*, 27(2):507–531, January 2005. https://doi.org/10.1137/040607605.

[95] Joel Robbin and Dietmar Salamon. *Introduction to Differential Geometry*. Springer, 2022. https://doi.org/10.1007/978-3-662-64340-2.

[96] Yousef Saad. *Numerical methods for large eigenvalue problems: revised edition*. SIAM, 2011.

[97] Hiroyuki Sato and Toshihiro Iwai. Optimization algorithms on the Grassmann manifold with application to matrix eigenvalue problems. *Japan Journal of Industrial and Applied Mathematics*, 31:355–400, 2014. https://doi.org/10.1007/s13160-014-0141-9.

[98] Peter H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966. https://doi.org/10.1007/BF02289451.

[99] Ohad Shamir. A stochastic pca and svd algorithm with an exponential convergence rate. In *International Conference on Machine Learning*, pages 144–152. PMLR, 2015.

[100] Ohad Shamir. Fast stochastic algorithms for svd and pca: Convergence properties and convexity. In *International Conference on Machine Learning*, pages 248–256. PMLR, 2016.

[101] Ohad Shamir, Nati Srebro, and Tong Zhang. Communication-efficient distributed optimization using an approximate newton-type method. In *International conference on machine learning*, pages 1000–1008. PMLR, 2014.

[102] Nian Shao and Wenbin Chen. Riemannian acceleration with preconditioning for symmetric eigenvalue problems. *arXiv preprint: 2309.05143*, 2023.

[103] Nian Shao, Wenbin Chen, and Zhaojun Bai. EPIC: a provable accelerated eigensolver based on preconditioning and implicit convexity. *SIAM J. Matrix Anal. Appl. (To appear)*, 2024.

[104] Steven T. Smith. Optimization techniques on Riemannian manifolds. In *Hamiltonian and gradient flows, algorithms and control*, volume 3 of *Fields Inst. Commun.*, pages 113–136. Amer. Math. Soc., Providence, RI, 1994. https://doi.org/10.1090/fic/003/09.

[105] Mahdi Soltanolkotabi, Adel Javanmard, and Jason D Lee. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*, 65(2):742–769, 2018. https://doi.org/10.1109/TIT.2018.2854560.

[106] Michael Spivak. *A Comprehensive Introduction to Differential Geometry*, volume 1 of *10*. Publish or perish, 2 edition, 1979.

[107] Andreas Stathopoulos and James R. McCombs. PRIMME: preconditioned iterative multimethod eigensolver—methods and software description. *ACM Trans. Math. Software*, 37(2), 2010. https://doi.org/10.1145/1731022.1731031.

[108] Ananda Theertha Suresh, Yu X. Felix, Sanjiv Kumar, and Brendan H. McMahan. Distributed mean estimation with limited communication. In *International Conference on Machine Learning*, pages 3329–3337. PMLR, 2017.

[109] Andrea Toselli and Olof Widlund. *Domain decomposition methods—algorithms and theory*, volume 34 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 2005. https://doi.org/10.1007/b137868.

[110] Lloyd N. Trefethen and David Bau, III. *Numerical linear algebra*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997. https://doi.org/10.1137/1.9780898719574.

[111] Constantin Udriste. *Convex functions and optimization methods on Riemannian manifolds*, volume 297. Springer Science & Business Media, 2013. https://doi.org/10.1007/978-94-015-8390-9.

[112] Melanie Weber and Suvrit Sra. Riemannian optimization via frank-wolfe methods. *Mathematical Programming*, 199(1):525–556, 2023. https://doi.org/10.1007/s10107-022-01840-5.

[113] Yung-Chow Wong. Sectional curvatures of Grassmann manifolds. *Proceedings of the National Academy of Sciences*, 60(1):75–79, May 1968. https://doi.org/10.1073/pnas.60.1.75.

[114] Jinchao Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34(4):581–613, 1992. https://doi.org/10.1137/1034116.

[115] Peng Xu, Bryan He, Christopher De Sa, Ioannis Mitliagkas, and Chris Re. Accelerated stochastic power iteration. In *International Conference on Artificial Intelligence and Statistics*, pages 58–67. PMLR, 2018.

[116] Zhiqiang Xu, Xin Cao, and Xin Gao. Convergence analysis of gradient descent for eigenvector computation. International Joint Conferences on Artificial Intelligence, 2018.

[117] Hongyi Zhang, Sashank J Reddi, and Suvrit Sra. Riemannian svrg: Fast stochastic optimization on Riemannian manifolds. *Advances in Neural Information Processing Systems*, 2016.

[118] Hongyi Zhang and Suvrit Sra. First-order methods for geodesically convex optimization. In *Conference on learning theory*, pages 1617–1638. PMLR, 2016.

[119] Hongyi Zhang and Suvrit Sra. An estimate sequence for geodesically convex optimization. In *Conference On Learning Theory*, pages 1703–1723. PMLR, 2018.

[120] Xingyu Zhou. On the fenchel duality between strong convexity and lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

[121] Yunkai Zhou, James R. Chelikowsky, and Yousef Saad. Chebyshev-filtered subspace iteration method free of sparse diagonalization for solving the kohn–sham equation. *Journal of Computational Physics*, 274:770–782, 2014. https://doi.org/10.1016/j.jcp.2014.06.056.

[122] Yunkai Zhou, Yousef Saad, Murilo L. Tiago, and James R. Chelikowsky. Parallel self-consistent-field calculations via Chebyshev-filtered subspace acceleration. *Phy. rev. E*, 74:066704, 2006. https://doi.org/10.1103/PhysRevE.74.066704.