Practical Accelerated Optimization on Riemannian Manifolds

Foivos Alimisis  Antonio Orvieto  Gary Bécigneul  Aurelien Lucchi
Department of Computer Science
ETH Zürich

Abstract

We develop a new Riemannian descent algorithm with an accelerated rate of convergence. We focus on functions that are geodesically convex or weakly-quasi-convex, which are weaker function classes compared to prior work that has considered geodesically strongly convex functions. Our proof of convergence relies on a novel estimate sequence which allows to demonstrate the dependency of the convergence rate on the curvature of the manifold. We validate our theoretical results empirically on several optimization problems defined on a sphere and on the manifold of positive definite matrices.

1. Introduction

The field of optimization plays a central role in machine learning. At its core is the problem of finding a minimum of a function \( f : H \rightarrow \mathbb{R} \). In the vast majority of applications in machine learning, \( H \) is considered to be a Euclidean vector space. However, a number of machine learning tasks can profit from a specialized problem-dependent Riemannian structure (Bonnabel, 2013; Zhang & Sra, 2016), which will be the focus of our discussion in this paper. Among the most popular types of methods to optimize \( f \) are first-order methods, such as gradient descent that simply updates a sequence of iterates \( \{ x_k \} \) by stepping in the opposite direction of the gradient \( \nabla f(x_k) \). In the case \( H = \mathbb{R}^n \), gradient descent as a first-order method has been shown to achieve a suboptimal convergence rate on convex problems. In a seminal paper, Nesterov (1983) showed that one can construct an optimal — i.e. accelerated — algorithm that achieves faster rates of convergence for both convex and strongly-convex functions. The convergence analysis of this algorithm relies heavily on the linear structure of \( H \) and it is not until recently that a first adaptation to Riemannian manifolds was derived by Zhang & Sra (2018). The algorithm by Zhang & Sra (2018) is shown to obtain an accelerated rate of convergence for functions that are known to be geodesically strongly-convex, provided that one initializes in a neighborhood of the (unique) solution. These functions are of particular interest as they might be non-convex in the Euclidean sense and they occur in some relevant computational tasks, such as the approximation of the Karcher mean of positive definite matrices (Zhang et al., 2016). However, many other interesting problems belong to the weaker class of geodesically convex functions, that includes problems defined on the cone of Hermitian positive definite matrices (Sra & Hosseini, 2015) which appear in various areas of machine learning such as tracking (Cheng & Vemuri, 2013) and medical imaging (Zhu et al., 2007).

In this paper, we therefore address the problem of deriving an algorithm that provably obtains an accelerated rate of convergence for functions that are geodesically convex but not necessarily strongly convex. We also consider the extension to the weaker class of geodesically weakly-quasi-convex objective functions. A more thorough motivation for investigating convex and weakly-quasi convex objectives in Riemannian optimization can be found in Section 4 of Alimisis et al. (2019). Our main contributions are:

1. We propose a new Riemannian algorithm which has an accelerated rate of convergence for geodesically convex and weakly-quasi-convex functions, up to a curvature-dependent neighborhood of the solution set. Our method is inspired from a recent work by Nesterov et al. (2018), and uses a small-dimensional relaxation (sDR) oracle (which can be solved approximately and in linear time) to perform adaptive linear coupling \(^1\) (Allen-Zhu & Orecchia, 2014). In order to provide theoretical guarantees for this new algorithm, we use a novel estimate sequence combined with advanced results from Riemannian geometry.

2. We prove that, even when the objective function is non-convex, our optimizer leads to a descent method (in contrast to the accelerated method proposed by Zhang & Sra (2018)) and that, in the worst case, it converges to a stationary point with the same rate as Riemannian Gradient Descent.

3. We validate our theoretical findings numerically on problems defined on manifolds of both positive curvature (Rayleigh quotient maximization) and negative curvature (operator scaling and Karcher mean approximation).

\(^1\)See discussion in the next section.
We show the empirical superiority of our method when compared to Riemannian algorithms designed for well-conditioned geodesically strongly-convex objectives.

2. Related Work

Accelerated Gradient Descent (AGD). The first accelerated gradient descent algorithm in Euclidean vector spaces is due to Nesterov (1983). Since then, the community has shown a deep interest in understanding the mechanism underlying acceleration. A recent trend has been to look at acceleration from a continuous-time viewpoint (Su et al. 2014; Wibisono et al. 2016). In this framework, AGD is seen as the discretization of a second-order ODE. Alternatively, Allen-Zhu & Orecchia (2014) showed how one can view (a more general version of) AGD as a primal-dual method performing linear coupling between gradient descent and mirror descent. Recently, Nesterov et al. (2018) proposed AGD’sDR, a modification of the method by Allen-Zhu & Orecchia (2014) which adaptively selects the linear coupling parameter (denoted by $\beta$) at each iteration using an approximate line search. This work will serve as an inspiration for us to design an accelerated Riemannian algorithm.

Riemannian optimization. Research in the field of Riemannian optimization has encountered a lot of interest in the last decade. A seminal book in the field is (Absil et al. 2009) which gives a comprehensive review of many standard optimization methods, but does not discuss acceleration. More recently, Zhang & Sra (2016) proved convergence rates for Riemannian gradient descent applied to geodesically convex functions. Acceleration in a Riemannian framework was first discussed by Liu et al. (2017), who claimed to have designed a Riemannian method with guaranteed acceleration. While their methodology is interesting, unfortunately, as discussed in (Zhang & Sra 2018), their algorithm relies on finding the exact solution to a nonlinear equation at each iteration, and it is not clear how difficult this additional problem might be or how approximation errors accumulate. Subsequently, Zhang & Sra (2018) developed the first computationally tractable accelerated algorithm on a Riemannian manifold, but their approach only has provable convergence for geodesically strongly-convex objectives (provided that one initializes sufficiently close to the solution). In contrast, we address the problem of achieving acceleration for the weaker classes of geodesically convex and weakly-quasi-convex objective functions, which is of significant practical interest (see discussion in Section 6). We note that extending the proof by Zhang & Sra (2018) to these weaker classes of functions is non straightforward due to some distortions between the tangent spaces of the sequence of iterates of the algorithm. Indeed, the estimate sequence used in Zhang & Sra (2018) relies on changing the tangent space at each step. These successive changes give rise to additional errors which can be tackled using the strong convexity assumption. However, we were unable to reproduce this proof for weaker function classes. Instead, we rely on an estimate sequence that is qualitatively different from the one used in (Zhang & Sra 2018) in order to avoid distortions produced by changing tangent spaces.

3. Background

3.1. Preliminaries from Differential Geometry

We review some basic notions from Riemannian geometry that are required in our analysis. For a full review, we refer the reader to some classical textbook.

Manifold. A differentiable manifold $M$ is a topological space that is locally Euclidean. This means that for any point $x \in M$, 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.

A Riemannian manifold $(M, g)$ is a differentiable manifold equipped with a Riemannian metric $g$, i.e. an inner product for each tangent space $T_x M$ at $x \in M$. We denote the inner product of $u, v \in T_x M$ 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 $\gamma : [0, 1] \to M$ of constant speed and of (locally) minimum length. They can be thought of as the Riemannian generalization of straight lines in Euclidean space. Geodesics are used to construct the exponential map $\exp_x : T_x M \to M$, defined by $\exp_x(v) = \gamma(1)$, where $\gamma$ is the unique geodesic such that $\gamma(0) = x$ and $\dot{\gamma}(0) = v$. The exponential map is locally a diffeomorphism. We denote the inverse of the exponential map (in the ball that it is defined) by $\log_x : M \to T_x M$. Geodesics also provide a way to transport vectors from one tangent space to another. This operation called parallel transport is usually denoted by $\Gamma^y_x : T_x M \to T_y M$.

Vector fields and covariant derivative. The correct 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 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. However, for the purpose of our discussion, we define $\log_{x_{k+1}}^a(x_k) - \log_{x_{k+1}}^b(x_k)$ appearing in the estimate sequence belong to different tangent spaces and are therefore not directly comparable (while they are exactly the same in the Euclidean case).

By “distortion”, we mean that when considering two successive iterates $x_k$ and $x_{k+1}$, the terms $\log_{x_k}(a) - \log_{x_k}(b)$ and
analysis, it will be sufficient to rely on a simple notion of covariant derivative that relies on the (more visualizable) notion of parallel transport. First, we define vector fields on a Riemannian manifold as sections of the tangent bundle.

**Definition 1.** Let $M$ be a Riemannian manifold. A vector field $X$ in $M$ is a smooth map $X : M \to T M$, where $T M$ is the tangent bundle, i.e. the collection of all tangent vectors in all tangent spaces of $M$, such that $p \circ X$ is the identity ($p$ is the projection from $T M$ to $M$).

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

**Definition 2.** Given two vector fields $X, Y$ in a Riemannian manifold $M$, we define the covariant derivative of $Y$ along $X$ to be

$$\nabla_X Y(p) = \lim_{h \to 0} \frac{\Gamma^h_{\gamma(0)} Y(\gamma(h)) - Y(p)}{h},$$

where $\gamma$ is the unique integral curve of $X$, starting from $p$.

### 3.2. Geodesic convexity

We remind the reader of the basic definitions needed in Riemannian optimization.

**Definition 3.** A subset $A \subseteq M$ of a Riemannian manifold $M$ is called geodesically uniquely convex, if every two points in $A$ are connected by a unique geodesic.

**Definition 4.** A function $f : A \to \mathbb{R}$ is called geodesically convex, if $f(\gamma(t)) \leq (1 - t)f(p) + tf(q)$, where $\gamma$ is the geodesic connecting $p, q \in M$.

Given a function $f : M \to \mathbb{R}$, the notions of differential and (Riemannian) inner product allow us to define the Riemannian gradient of $f$ at $x \in M$, which is a tangent vector belonging to the tangent space based at $x, T_x M$.

**Definition 5.** The Riemannian gradient $\nabla_x f$ of a (real-valued) function $f : M \to \mathbb{R}$ at a point $x \in M$, is the tangent vector at $x$, such that $\langle \nabla_x f, u \rangle = df(x)[u]$ for any $u \in T_x M$.

Given the notion of Riemannian gradient and covariant derivative we can define the notion of Riemannian Hessian.

**Definition 6.** Given vector fields $X, Y$ in $M$, we define the Hessian operator of $f$ to be

$$\text{Hess}(f)(X, Y) = \langle \nabla_X \nabla f, Y \rangle.$$

Using the Riemannian inner product and the Riemannian gradient, we can formulate an equivalent definition for geodesic convexity for a smooth function $f$ defined in a geodesically uniquely convex domain $A$.

**Proposition 1.** Let a smooth, geodesically convex function $f : A \to \mathbb{R}$. Then, for any $x, y \in A$, we have

$$f(x) - f(y) \geq \langle \nabla f(y), \log_x y \rangle$$

As in the Euclidean case, any local minimum of a geodesically convex function is a global minimum. We now generalize the well-known notion of Euclidean weak-quasi-convexity to Riemannian manifolds. For a review of the notion, we refer the reader to (Guminov & Gasnikov, 2017).

**Definition 7.** A function $f : A \to \mathbb{R}$ is called geodesically $\alpha$-weakly-quasi-convex with respect to $c \in M$, if

$$\alpha(f(x) - f(c)) \leq -\langle \nabla f(x), \log_x c \rangle$$

for some fixed $\alpha \in (0, 1]$ and any $x \in M$.

It is easy to see that weak-quasi-convexity implies that any local minimum of $f$ is also a global minimum. Using the notion of parallel transport we can define when $f$ is geodesically $L$-smooth, i.e. has Lipschitz continuous gradient in a suitable differential-geometric way.

**Definition 8.** A function $f : A \to \mathbb{R}$ is called $L$-smooth if

$$\| \nabla f(x) - \Gamma^a_y \nabla f(y) \| \leq L \| \log_x y \|$$

for any $x, y \in M$.

Geodesic $L$-smoothness has similar properties to its Euclidean analogue. Namely, a two times differentiable function is $L$-smooth, if and only if the norm of its Riemannian Hessian is bounded by $L$. Also if a function $f$ is $L$-smooth, we have that, for any $x, y \in M$,

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

### 3.3. Basic Assumptions

In this paper, we make the standard assumption that the input space is not "infinitely curved". In order to make this statement rigorous, we need the notion of sectional curvature $K$, which is a measure of how sharply the manifold is curved (or how "far" from being flat our manifold is), "two-dimensionally". More concretely, as in (Zhang & Sra, 2018), we make the following set of assumptions:

**Assumption 1.** Given $A \subseteq M$ geodesically uniquely convex, and $f : A \to \mathbb{R}$,

1. The sectional curvature $K$ inside $A$ is bounded from above and below, i.e. $K_{\min} \leq K \leq K_{\max}$.
2. $A$ is a geodesically uniquely convex subset of $M$, such that diam$(A) \leq D$. The exponential map $\exp_x$ is globally a diffeomorphism for any $x \in A$ with inverse denoted by $\log_x$.
3. $f$ is geodesically $L$-smooth with all minima inside $A$. 
4. The RAGDsDR Algorithm

We now develop a Riemannian accelerated algorithm inspired by the Euclidean algorithm presented in [Nesterov et al., 2018]. This algorithm is detailed in Algorithm 1 and illustrated in Figure 1. Each iteration $k$ is similar to the Euclidean algorithm since the next iterate $x_{k+1}$ (line 4) is computed by taking a gradient step at an interpolated point $y_k$ (line 3) which follows the direction of a momentum term $\log_{y_k}(x_k)$. The main difference is that the curve $y_k$ to $x_k$ is a geodesic on the manifold $M$ instead of a straight Euclidean line. As in [Nesterov et al., 2018], we also rely on a minimization over a closed interval (i.e. the small-dimension relaxation, sDR) to choose the best possible stepsizes $\beta_k$ (line 2) on the geodesic connecting $y_k$ to $x_k$. We will see in the next section that this minimization is computationally fast to solve (also see Section 6), can be computed approximately and practically yields faster convergence than the typical fixed parameter $\beta_k = \frac{1}{k+2}$ (Nesterov, 2018).

Algorithm 1 RAGDsDR for convex functions

1. $A_0 = 0, x_0 = u_0, v_0(x) = \frac{1}{2} \| \log_{y_k}(x) \|^2$
2. $\beta_k = \text{argmin}_{\beta \in [0,1]} f(\exp_{y_k}(\beta \log_{y_k}(x_k)))$
3. $y_k = \exp_{y_k}(\beta_k \log_{y_k}(x_k))$
4. $x_{k+1} = \exp_{y_k}(-\frac{1}{2} \nabla f(y_k))$
5. $\zeta_{k+1} = \frac{1}{2}$
6. $A_{k+1} = A_k + a_{k+1}$
7. $v_{k+1} = \exp_{v_k}(-a_{k+1} \Gamma_{y_k} \nabla f(y_k))$

The first condition follows by setting $\beta = 1$ in step 2. For the second condition, we consider different cases depending on the value of $\beta$. We have to take into consideration that $y_k$ is on the geodesic connecting $v_k$ with $x_k$. The derivative of the curve $\exp_{v_k}(\beta \log_{v_k}(x_k))$ with respect to $\beta$ is tangent to the geodesic and has length equal to $\| \log_{v_k}(x_k) \|$, because geodesics have constant speed. This means that the derivative at the point $y_k$ is equal to $\Gamma_{y_k} \log_{v_k}(x_k)$. By relying on the optimality condition of $\beta$, we distinguish the following three cases:

(i) If $\beta_k = 0$, then $\langle \nabla f(y_k), \Gamma_{y_k} \log_{y_k}(x_k) \rangle \geq 0$ and $y_k = v_k$, thus $\langle \nabla f(y_k), \log_{v_k}(y_k) \rangle = 0$.

(ii) If $\beta_k \in (0, 1)$, then $\langle \nabla f(y_k), \Gamma_{y_k} \log_{v_k}(x_k) \rangle = 0$ and $\log_{v_k}(y_k) = \beta_k \log_{y_k}(x_k)$. Thus, $\langle \nabla f(y_k), \frac{1}{2k} \Gamma_{y_k} \log_{v_k}(y_k) \rangle = 0$, which implies $\langle \nabla f(y_k), \log_{v_k}(y_k) \rangle = 0$.

(iii) If $\beta_k = 1$, then $\langle \nabla f(y_k), \Gamma_{y_k} \log_{v_k}(x_k) \rangle \leq 0$ and $y_k = x_k$.

We deduce that $\langle \nabla f(y_k), \Gamma_{y_k} \log_{v_k}(y_k) \rangle \leq 0$, thus $\langle \nabla f(y_k), \log_{v_k}(y_k) \rangle \leq 0$ and finally $\langle \nabla f(y_k), \log_{v_k}(y_k) \rangle \geq 0$.

In a practical setting, the line search procedure is inexact. While we can still expect the first inequality in Eq. 1 to be satisfied exactly, the second one can only be satisfied up to a small error $\varepsilon$, i.e. $\langle \nabla f(y_k), \log_{v_k}(y_k) \rangle \geq -\varepsilon$. We note that this is an analog condition to the one used by [Nesterov et al., 2018] in the Euclidean case.

As we will see shortly, one of the main quantities of interest in our analysis will be

$$E_k(x) := \langle \nabla f(y_k), \log_{v_k}(x) - \Gamma_{y_k} \log_{v_k}(x) \rangle$$

We will prove that this error is lower bounded by two terms, namely $E_k(x) \geq -\varepsilon - \eta = -\varepsilon$, where $\varepsilon$ is the error obtained by the line search and $\eta$ is an extra curvature-dependent error, which depends on a bound $D$ for the working domain. This will be made precise in the next section.

4 We use Fermat’s theorem for $f(\exp_{v_k}(\beta \log_{v_k}(y_k)))$.
5 $f(\exp_{v_k}(\beta \log_{v_k}(y_k)))$ is locally decreasing on the left.
5. Proof of Convergence

5.1. Geodesically-convex functions

We first derive a lemma proving an accelerated rate to \( x^* = \text{argmin}_x f(x) \) up to a ball whose size is defined by the error term \( \mathcal{E}_k(x^*) \geq -\theta \) (see Eq. 1) at \( x^* \). Afterwards, we further discuss how the error \( \theta \) relates to the properties of the manifold \( M \) where the function \( f \) is defined.

**Lemma 2.** Assume that our working domain \( A \) has diameter bounded by \( D \). If \( f : A \rightarrow R \) is a geodesically convex and \( L \)-smooth function, the iterates \( \{x_k\} \) produced by Algorithm 1 obtain the following accelerated rate:

\[
\frac{d}{dt} \Gamma_{X(t)}^{y_k} \log X(t)(x) = \Gamma_{X(t)}^{y_k} \nabla X(t) \log X(t)(x),
\]

where \( \nabla X \) is the covariant derivative along \( X \) as defined in Def. 2. Now we have that

\[
\nabla X \log X(t)(x) = \nabla X (\frac{1}{2} d(X,x)^2(t)) = \nabla X \left( \frac{1}{2} d(X,x)^2 \right)(t) = \text{Hess}_X \left( \frac{1}{2} d(X,x)^2 \right)(t).
\]

The derivation of the second equality can be found in (Lee 2018), Chapter 11. The last equality holds because the Hessian is by definition equal to \( \nabla \text{grad} \), and since \( X \) is a geodesic, we have: \( X(t) = \Gamma_{y_k}^{X(t)} \log y_k(v_k) \).

The error term \( \mathcal{E}_k(x) \) is defined as

\[
\mathcal{E}_k(x) = \langle \text{grad} f(y_k), \log y_k(x) - \Gamma_{v_k}^{y_k} \log v_k(x) \rangle, 
\]

where \( \log y_k \) is the minimum of the function \( \psi_k \), which will be constructed by taking a linear combination of lower bounds on the function \( f \) constructed at each iterate where the function is being evaluated. We wish to prove that with a suitable choice of \( \psi_k^* \), we have that:

**C1** \( A_k f(x_k) \leq \psi_k^* \) (see definition of \( A_k \) in Algorithm 1)

**C2** At least for \( x = x^* \),

\[
\psi_{k+1}(x) \leq \psi_k(x) + a_{k+1}(f(y_k) + \langle \text{grad} f(y_k), \log y_k(x) \rangle + \bar{\theta}).
\]

As detailed in the appendix, combining C1 and C2 yields the desired result.

**Proof sketch.** As in (Nesterov et al. 2018), the proof relies on an estimate sequence of functions, defined as

\[
\psi_k(x) = \psi_k^* + \frac{1}{2} \| \log v_k(x) \|^2,
\]

where \( \psi_k^* \) is the minimum of the function \( \psi_k \), which will be constructed by taking a linear combination of lower bounds on the function \( f \) constructed at each iterate where the function is being evaluated. We wish to prove that with a suitable choice of \( \psi_k^* \), we have that:

**C1** \( A_k f(x_k) \leq \psi_k^* \) (see definition of \( A_k \) in Algorithm 1)

**C2** At least for \( x = x^* \),

\[
\psi_{k+1}(x) \leq \psi_k(x) + a_{k+1}(f(y_k) + \langle \text{grad} f(y_k), \log y_k(x) \rangle + \bar{\theta}).
\]

As detailed in the appendix, combining C1 and C2 yields the desired result. □

**Control of the Error \( \bar{\theta} \).** We now further investigate the error term \( \bar{\theta} \) that appears in Lemma 2. Consider the function \( g : [0,1] \rightarrow R \) defined as

\[
g(t) = \langle \text{grad} f(y_k), \Gamma_{X(t)}^{y_k} \log X(t)(x) \rangle,
\]

where \( X : [0,1] \rightarrow M \) is the geodesic connecting \( y_k = X(0) \) and \( v_k = X(1) \). By the mean value theorem, there exists some \( t_0 \in (0,1) \), such that \( g(1) - g(0) = g(t_0) \). This is equivalent to

\[
\mathcal{E}_k(x) = \langle \text{grad} f(y_k), \log y_k(x) - \Gamma_{v_k}^{y_k} \log v_k(x) \rangle
\]

\[
= \langle \text{grad} f(y_k), \frac{d}{dt} \bigg|_{t=t_0} \Gamma_{X(t)}^{y_k} \log X(t)(x) \rangle
\]

\[
= \langle \text{grad} f(y_k), -\Gamma_{X(t_0)}^{y_k} \nabla X(t_0) \log X(t)(x) \rangle_{t=t_0}.
\]

The last equality holds because of a known property of parallel transport:

\[
\frac{d}{dt} \Gamma_{X(t)}^{y_k} \log X(t)(x) = \Gamma_{X(t)}^{y_k} \nabla X(t) \log X(t)(x),
\]

while the smallest eigenvalue is lower bounded by

\[
\text{Hess}_X \left( \frac{1}{2} d(X,x)^2 \right) = \text{Hess}_X \left( \frac{1}{2} d(X,x)^2 \right)(t) = \text{Hess}_X \left( \frac{1}{2} d(X,x)^2 \right)(t).
\]

The eigenvalues of the operator \( \mathcal{H} \) are exactly equal to the ones of \( \text{Hess}_X \left( \frac{1}{2} d(X,x)^2 \right) \), because \( \Gamma_{X(t)}^{y_k} = \Gamma_{X(t)}^{y_k} \left( \Gamma_{X(t)}^{y_k} \right)^{-1} \), thus the norm of the operator \( \mathcal{H} - I_d \) satisfies

\[
\| \mathcal{H} - I_d \| \leq \max \{ \zeta - 1, 1 - \delta \}
\]

Now, observe that the quantity \( \mathcal{E}_k(x) \) can be manipulated as

\[
\text{Recall that the geodesic } X, \text{ defined as } X(t) = \text{exp}(t \cdot \log y_k(v_k)), \text{ has constant velocity and the parallel transport of a tangent vector along } X \text{ remain tangent. Thus transporting parallelly } \log y_k(v_k) = X(0) \text{ from } X(0) \text{ to } X(t) \text{ gives the velocity at } X(t),\text{ i.e. } \dot{X}(t).
\]
follows,
\[ \mathcal{E}_k(x) = \langle \nabla f(y_k), \log y_k(x) - \Gamma y_k \log y_k(x) - \log y_k(v_k) \rangle + \langle \nabla f(y_k), \log y_k(v_k) \rangle \]
\[ \geq \langle \nabla f(y_k), \log y_k(x) - \Gamma y_k \log y_k(x) - \log y_k(v_k) \rangle - \tilde{\epsilon}, \]
where the last inequality holds by definition of \( \tilde{\epsilon} \) (by the line search) which is such that \( \langle \nabla f(y_k), \log y_k(v_k) \rangle \geq -\tilde{\epsilon} \).

Using Eq. \ref{eq:5.1} we finally get
\[ \langle \nabla f(y_k), \log y_k(x) - \Gamma y_k \log y_k(x) - \log y_k(v_k) \rangle \]
\[ \geq -\left\| \nabla f(y_k) \right\|_\mathcal{H} - L_d \left\| \log y_k \right\| \]
\[ \geq -LD \max \{\zeta - 1, 1 - \delta\} D \]
\[ = -LD^2 \max \{\zeta - 1, 1 - \delta\}. \]

To conclude, we obtain the following corollary with an explicit form for the error \( \hat{\theta} = \tilde{\epsilon} + \tilde{\eta} \) as follows.

**Theorem 3.** Consider a Riemannian manifold \( M \) with curvature \( K \) bounded from above and below, \( K_{\min} \leq K \leq K_{\max} \). Assume that our working domain \( A \) has diameter bounded by \( D \). If \( f : A \to \mathbb{R} \) is a geodesically convex and \( \mathcal{L} \)-smooth function, the iterates \( \{x_k\} \) produced by Algorithm 1 obey the following accelerated rate:

\[ f(x_k) - f(x^*) \leq \frac{2L^2D^2}{k^2} + \frac{L^2D^2 \max \{\zeta - 1, 1 - \delta\} + \epsilon}{\eta} \]
\[ = LD^2 \left( \frac{2\zeta}{k^2} + \max \{\zeta - 1, 1 - \delta\} \right) + \hat{\theta} \]

where \( \zeta = \begin{cases} \sqrt{-k_{\min}} D \coth \left( \sqrt{-k_{\min}} D \right), & k_{\min} < 0 \\ 1, & k_{\min} \geq 0 \end{cases} \)

and \( \delta = \begin{cases} 1, & k_{\max} \leq 0 \\ \sqrt{k_{\max}} D \cot \left( \sqrt{k_{\max}} D \right), & k_{\max} > 0 \end{cases} \)

5.2. Geodesically weakly-quasi-convex functions

We extend Algorithm 1 to functions that are \( \alpha \)-weakly-quasi-convex. This requires to restart Algorithm 1 whenever the suboptimality is less than the previous one by a factor \( 1 - \frac{c}{\alpha} \), where \( c > 1 \) is a constant.

**Theorem 4.** Algorithm 2 applied to an \( \alpha \)-weakly-quasi-convex function produces a sequence of iterates \( \{x_k\}_{k=1}^N \) such that

\[ f(x_N) - f(x^*) \leq O \left( \frac{L^2D^2}{\alpha^2N^2} \right) + \frac{c}{(c - 1)\alpha} \hat{\theta}, \]

where \( \hat{\theta} \) is the same error as in the convex case and \( c > 1 \).

As in the convex case, we have \( \hat{\theta} \leq L^2 \max \{\zeta - 1, 1 - \delta\} + \tilde{\epsilon}. \) When \( c \to 0, d \to \infty \) and \( M \to \frac{1}{\alpha} \hat{\theta} \), we almost recover the convex case for \( \alpha = 1 \).

**Algorithm 2 RAGDsDR for weakly-quasi-convex functions**

1. for \( i \geq 0 \) do
2. \( A_0 = 0, x_0 = u_0, \psi_0(x) = \frac{1}{2} \left\| \log u_0 \right\|^2 \)
3. for \( k \geq 0 \) do
4. \( \beta_k = \arg \min_{\beta \in [0,1]} \{ f(\exp_{v_k} (\beta \log y_k(x_k))) \} \)
5. \( y_k = \exp_{v_k} (\beta \log y_k(x_k)) \)
6. \( x_{k+1} = \exp_{v_k} (-\frac{1}{L} \nabla f(y_k)) \)
7. \( \gamma_{k+1} = \frac{\beta_k}{\alpha_k + \beta_k} = \frac{1}{\alpha} \)
8. \( A_{k+1} = A_k + \beta_k \)
9. \( v_{k+1} = \exp_{v_k} (-\alpha_k + \beta_k \log y_k(x_k)) \)
10. if \( f(x_k^*) - f(x^*) \leq (1 - \frac{c}{\alpha}) (f(x_0^*) - f(x^*)) \) then
11. break
12. end if
13. \( k = k + 1 \)
14. end for
15. \( x_{i+1} = x_N \) (where \( N \) is the number of steps performed in the loop over \( k \))
16. \( i = i + 1 \)
17. end for

5.3. Geodesically non-convex functions

We conclude the section by proving that Algorithm 1 has some basic convergence properties in the non-convex case.

**Proposition 5.** Algorithm 1 applied to an \( \mathcal{L} \)-smooth objective function \( f \) for \( N \)-many steps, produces iterates \( \{y_k\} \), such that

\[ \min_{k=0,\ldots,N} \left\| \nabla f(y_k) \right\|^2 \leq \frac{2L(f(x_0) - f(x^*))}{N}. \]

This result matches the convergence rate of Riemannian Gradient Descent in non-convex smooth optimization. This follows directly from the conditions in Equation \ref{eq:1}.

6. Numerical Experiments

We validate our findings on Riemannian manifolds of both positive and negative curvature. Our code is built on top of Pymanopt \cite{Townsend2016}. We compare RAGDsDR (Algorithm 1) with Riemannian Gradient Descent (RGD) and, when possible (i.e. when we can estimate the strong convexity modulus), with RAGD by Zhang & Sra \cite{Zhang2018}. As a more practical alternative to the line search in step 2 (which we solve with 10 iterations of golden-section search), we show the performance for \( \beta_k = \frac{k}{k+2} \). Under this choice, RAGDsDR recovers a Riemannian version of Linear Coupling \cite{Allen-Zhu2014}.

\[ \text{https://github.com/aorvieto/RAGDsDR.git} \]
6.1. Positive curvature

We first consider the problem of maximizing the Rayleigh quotient \( \frac{x^TAx}{2\|x\|^2} \) over \( \mathbb{R}^d \), i.e., of finding the dominant eigenvector of \( A \in \mathbb{R}^{d \times d} \). This non-convex problem can be written on the open hemisphere \( \mathbb{S}^{d-1} \) (\( \mathbb{S}^{d-1} \) is a manifold of constant positive curvature): \( \arg\min_{x \in \mathbb{S}^{d-1}} f(x) := -\frac{1}{2} x^T Ax \). It is well known that, in the Euclidean case, such an objective is hard to optimize if \( A \) is high-dimensional and ill-conditioned — and is therefore able to truly showcase the acceleration phenomenon\(^9\) for convex but not necessarily strongly-convex functions, in a tight way.

Setup details. We choose \( A = \frac{1}{2} B B^T \), where \( B \in \mathbb{R}^{d \times n} \) has standard Gaussian entries\(^8\). To make the problem computationally interesting, we choose \( d = 1000 \) and \( n = 1050 \approx d \), leading to a large condition number. In correspondence to the Euclidean case, we have \( L = \lambda_{\max}(A) \) and use a step-size of \( 1/L \) for RGD and RAGD. Also, we choose the strong-convexity modulus \( \mu \) (needed parameter for RAGD) as \( \lambda_{\min}(A) \), again in correspondence with the Euclidean case.

Results. As predicted by Theorem 3, Figure 2 shows that RAGDsDR is able to accelerate RGD from \( O(1/k) \) to \( O(1/k^2) \) during the first hundred iterations. The rate will eventually\(^10\) become linear, due to the gradient-dominance of \( f \) (Theorem 4 in Zhang et al. (2016)). Similarly, RAGD is only able to profit from acceleration at a late stage — and before then it is comparable to RGD.

Finally, we note that the choice \( \beta_k = \frac{k}{k+2} \), which reduces by a factor of 10 the iteration-cost of RAGDsDR, does not influence much the empirical rate. Indeed, as shown in the figure, line search returns a result which is somehow similar. However, as also mentioned in Nesterov et al. (2018), the line search clearly increases the adaptiveness of the method to curvature, providing better stability (no oscillations) and steady function decrease at each iteration — even for non-convex potentials (see Proposition 5).

6.2. Negative curvature

We now consider two problems on the space of \( d \times d \) real symmetric positive definite matrices \( \mathcal{S}^{++} \). The Riemannian metric \( g_A(M, N) = \text{trace}(A^{-1}MA^{-1}N) \) makes \( \mathcal{S}^{++} \) a complete Riemannian manifold with negative curvature \((\text{Bhatia}, 2009)\).

Operator Scaling. Consider an operator \( T : \mathbb{R}^{d \times d} \rightarrow \mathbb{R}^{d \times d} \) defined by an \( m \)-tuple of \( d \times d \) matrices \( (A_j)_{j=1}^m \): \( T(X) = \sum_{j=1}^m A_j X A_j^T \). The problem of operator scaling consists in finding \( n \times n \) matrices \( X \) and \( Y \) such that if \( A_j := Y^{-1}A_jX \), then \( \sum_{j=1}^m A_j \) becomes positive-definite to order \( \sqrt{m} \). Such problem is of extreme interest in theoretical computer science (Garg et al., 2018), and has applications in algebraic complexity, invariant theory, analysis and quantum information. Gurvits (2004) showed that one can solve operator scaling by computing the capacity of \( T \), i.e., by finding \( \arg\min_{X \in \mathbb{S}^{++}(d)} \frac{\det(T(X))}{\det(X)} \). This function is non-convex in \( \mathbb{R}^{d \times d} \), but its logarithm\(^11\) is geodesically convex on \( \mathcal{S}^{++} \)\((\text{Vishnoi}, 2018)\). Recently, Allen-Zhu et al. (2018) were able to exploit this property to design a competitive second-order Riemannian optimizer to solve operator scaling. Here, we instead test the performance of accelerated first-order methods.

To the best of our knowledge, there does not exist any estimate of the strong convexity constant for the log-capacity. Hence, RAGD (Zhang & Sra, 2018) is not applicable to operator scaling. Instead, we compare the performance of RAGsDR with the iterative solution by Gurvits (2004) in Figure 3, showing again a significant acceleration.

Karcher mean. Given an \( n \)-tuple of \( d \times d \) positive definite matrices \( (A_j)_{j=1}^n \), the Karcher mean is the unique positive definite solution \( X \) to the equation \( \sum_{i=1}^n \log(A_i^{-1} X) \), where \( \log \) is the matrix logarithm. This matrix average has many desirable properties (see exhaustive list in Ando et al., 2004), which make its fast computation relevant to signal processing and medical imaging. It turns out that the Karcher mean can be written as \( \arg\min_{X \in \mathcal{S}^{++}(d)} \text{log}(X) = \frac{1}{m} \sum_{i=1}^m d(A_i, X)^2 \). Clearly, \( f \) is strongly-convex with modulus \( \mu = 1 \), and \( L \)-smooth with modulus estimated to be around 5 \((\text{Zhang & Sra}, 2016)\). Following Zhang & Sra...
Figure 3. Scaling of a positive operator by minimizing its log-capacity. Shown is the distance to double stochasticity (Def. 2.9 from [Garg et al., 2018]). In this metric, RAGDsDR is not necessarily a descent method. Here we estimate $L = 1$ (the smallest value that guarantees numerical stability), and note that the algorithm by [Gurvits, 2004] is very similar to RGD with step $1/L$. The rate appears to be sublinear (yet faster that $O(1/k^2)$), in accordance with the complexity result in [Garg et al., 2018].

In Figure 4 we show that RAGDsDR (with line search) is able to achieve a faster rate compared to RAGD in terms of number of iterations. Interestingly, here the choice $\beta_k = k + 2$ only leads to a slight initial acceleration compared to RGD. This behavior can be explained by looking at the values of $\beta_k$ returned by line search: for the first iterations $\beta_k$ is set to a very small value — leading to convergence in 10 iterations.

7. Discussion

We proposed novel algorithms for the minimization of geodesically convex and weakly-quasi-convex real-valued functions defined on a Riemannian manifold of bounded sectional curvature. We derived theoretical guarantees proving these algorithms achieve accelerated rates of convergence and validated our results empirically. We conclude by contrasting our results to prior work and discussing further extensions.

Extension to strongly-convex case. Extending our analysis to the strongly-convex case appears non-trivial. Existing analyses such as [Zhang & Sra, 2018] that consider such functions have an extra term $\frac{\sqrt{\delta}}{\sqrt{\kappa}} \|y_k\|_2^2$ appearing in the estimate sequence, which cannot straightforwardly be dealt with in our current proof.

Euclidean setting. Our result recovers the Euclidean case for $\zeta = \delta = 1$. Indeed, as the curvature of the manifold tends to 0, $\zeta$ and $\delta$ tend to 1, in which case the extra error $\hat{\theta}$ tends to 0. Alternatively, the error $\hat{\theta}$ can be decreased by further restricting the diameter of the domain $D$. We should mention that the proposed bound for the quantity $\hat{\eta}$ is rather conservative.

Initialization used in [Zhang & Sra, 2018]. Theorem 3 in [Zhang & Sra, 2018] relies on the restrictive assumption that the initialization of their algorithm is restricted to a ball of radius $D = \frac{1}{2\sqrt{\kappa R}} \left( \frac{\zeta}{\delta} \right)^2$ centered at $x^*$. Using the strong convexity of the objective function, they are able to prove that the working domain is expanded until $\frac{1}{10\sqrt{R}} \left( \frac{\zeta}{\delta} \right)^2 \leq \frac{1}{10\sqrt{R}}$. Given that we do not use strong convexity (but just convexity), this assumption would translate to a bound on the working domain of $D \leq \frac{1}{10\sqrt{R}}$. This would in turn imply $\zeta \approx 1.003$ and $\delta \approx 0.997$, for which the extra error $\hat{\theta}$ is close to $0.003\kappa D^2$ and the final upper bound of the rate of convergence is

$$f(x_k) - f(x^*) \leq \kappa \left( \frac{2C}{K^2} + 0.003 \right).$$

Even in the worst-case, the quantity 0.003 is relatively small compared to $\frac{2C}{K^2}$, and one can therefore perform many steps of the algorithm without the extra error affecting the rate.

Further improvements. One question of practical relevance surrounds the extra error term $\hat{\theta}$ which appears in both Theorem 3 and 4 but was not observed in our empirical results. Our analysis reveals that it is a continuous function of how far the manifold $M$ is from being Euclidean. This can be seen clearly in the expression $\hat{\theta} = \epsilon + \kappa D^2 \max \{ \zeta - 1, 1 - \delta \}$. One problem that goes beyond the scope of our work is to determine whether such term is an artifact of our worst-case analysis. Alternatively, an interesting direction would be to study whether the extra error arises as the numerical discretization error of the ODE.
Practical Accelerated Optimization on Riemannian Manifolds

derived in [Alimisis et al., 2019]. Practically, this error is however not a significant problem since one can switch to a non-accelerated method once we reach the $\tilde{\theta}$ level set.

Acknowledgements The authors would like to thank professors Nicolas Boumal and Suvrit Sra for helpful discussions regarding the content of this work. Gary Bécigneul is funded by the Max Planck ETH Center for Learning Systems.

References

Absil, P.-A., Mahony, R., and Sepulchre, R. Optimization algorithms on matrix manifolds. Princeton University Press, 2009.

Alimisis, F., Orvieto, A., Bcigneul, G., and Lucchi, A. A continuous-time perspective for modeling acceleration in riemannian optimization, 2019.

Allen-Zhu, Z. and Orecchia, L. Linear coupling: An ultimate unification of gradient and mirror descent. arXiv preprint arXiv:1407.1537, 2014.

Allen-Zhu, Z., Garg, A., Li, Y., Oliveira, R., and Wdigerson, A. Operator scaling via geodesically convex optimization, invariant theory and polynomial identity testing. In Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, pp. 172–181, 2018.

Ando, T., Li, C.-K., and Mathias, R. Geometric means. Linear algebra and its applications, 385:305–334, 2004.

Bhatia, R. Positive definite matrices, volume 24. Princeton university press, 2009.

Bini, D. A. and Iannazzo, B. Computing the karcher mean of symmetric positive definite matrices. Linear Algebra and its Applications, 438(4):1700–1710, 2013.

Bonnabel, S. Stochastic gradient descent on riemannian manifolds. IEEE Transactions on Automatic Control, 58 (9):2217–2229, 2013.

Cheng, G. and Vemuri, B. C. A novel dynamic system in the space of spd matrices with applications to appearance tracking. SIAM journal on imaging sciences, 6(1):592–615, 2013.

Garg, A., Gurvits, L., Oliveira, R., and Wdigerson, A. Algorithmic and optimization aspects of brascamp-lieb inequalities, via operator scaling. Geometric and Functional Analysis, 28(1):100–145, 2018.

Guminov, S. and Gasnikov, A. Accelerated methods for $\alpha$-weakly-quasi-convex problems. arXiv preprint arXiv:1710.00797, 2017.

Gurvits, L. Classical complexity and quantum entanglement. Journal of Computer and System Sciences, 69(3):448–484, 2004.

Lee, J. M. Introduction to Riemannian manifolds, volume 176. Springer, 2018.

Liu, Y., Shang, F., Cheng, J., Cheng, H., and Jiao, L. Accelerated first-order methods for geodesically convex optimization on riemannian manifolds. In Advances in Neural Information Processing Systems, pp. 4868–4877, 2017.

Nesterov, Y. Lectures on convex optimization, volume 137. Springer, 2018.

Nesterov, Y., Gasnikov, A., Guminov, S., and Dvurechensky, P. Primal-dual accelerated gradient descent with line search for convex and nonconvex optimization problems. arXiv preprint arXiv:1809.05895, 2018.

Nesterov, Y. E. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. In Dokl. akad. nauk Sssr, volume 269, pp. 543–547, 1983.

Sra, S. and Hosseini, R. Conic geometric optimization on the manifold of positive definite matrices. SIAM Journal on Optimization, 25(1):713–739, 2015.

Su, W., Boyd, S., and Candes, E. A differential equation for modeling nesterovs accelerated gradient method: Theory and insights. In Advances in Neural Information Processing Systems, pp. 2510–2518, 2014.

Townsend, J., Koep, N., and Weichwald, S. Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation. The Journal of Machine Learning Research, 17(1):4755–4759, 2016.

Vishnoi, N. K. Geodesic convex optimization: Differentiation on manifolds, geodesics, and convexity. arXiv preprint arXiv:1806.06373, 2018.

Wibisono, A., Wilson, A. C., and Jordan, M. I. A variational perspective on accelerated methods in optimization. proceedings of the National Academy of Sciences, 113(47):E7351–E7358, 2016.

Zhang, H. and Sra, S. First-order methods for geodesically convex optimization. In Conference on Learning Theory, pp. 1617–1638, 2016.

Zhang, H. and Sra, S. Towards riemannian accelerated gradient methods. arXiv preprint arXiv:1806.02812, 2018.

Zhang, H., Reddi, S. J., and Sra, S. Riemannian svrg: Fast stochastic optimization on riemannian manifolds. In Advances in Neural Information Processing Systems, pp. 4592–4600, 2016.
Zhu, H., Zhang, H., Ibrahim, J. G., and Peterson, B. S. Statistical analysis of diffusion tensors in diffusion-weighted magnetic resonance imaging data. *Journal of the American Statistical Association*, 102(480):1085–1102, 2007.
A. The operator $H$

An important operator in the control of the extra error $\theta$ arising due to the "jump" we do in our estimate sequence is $H = -\Gamma^y_{X(t)} \text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)\Gamma^y_{y_k} : T_{y_k}M \to T_{y_k}M$. This is actually a whole family of operators depending on $t$. Let us fix some $t$, i.e. fix one operator of the family.

- The eigenvalues of $H$ are equal to the eigenvalues of $-\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)$. Indeed, the operator $-\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)$ is diagonalizable (check [Alimisis et al., 2019]) and can be written as $UDU^{-1}$ in a unique way, where $D$ is diagonal formed by its eigenvalues and $U$ by its eigenvectors. Then the operator $H$ has a unique representation in the form $\Gamma^y_{X(t)} UDU^{-1}(\Gamma^y_{X(t)})^{-1} = (\Gamma^y_{X(t)} U) D(\Gamma^y_{X(t)} U)^{-1}$ and its eigenvalues are the diagonal entries of $D$.

- The largest eigenvalue of $-\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)$ is less or equal than
  \[ \zeta = \begin{cases} \sqrt{-K_{\text{min}}d(X, x^*)} \coth(\sqrt{-K_{\text{min}}d(X, x^*)}) & , K_{\text{min}} < 0 \\ 1 & , K_{\text{min}} \geq 0 \end{cases} \]
  
  and the smallest more or equal than
  \[ \delta = \begin{cases} 1 \sqrt{K_{\text{max}}d(X, x^*)} \cot(\sqrt{K_{\text{max}}d(X, x^*)}) & , K_{\text{max}} \leq 0 \\ \delta & , K_{\text{max}} > 0 \end{cases} \]
  
  Indeed, Lemma 2 in [Alimisis et al., 2019] implies that
  \[ \delta \| \dot{X} \|^2 \leq \langle -\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)\dot{X}, \dot{X} \rangle \leq \zeta \| \dot{X} \|^2 \]
  
  for any curve $X$. Thus for a vector $v \in T_{X(t)}M$ we can choose a curve $\dot{X}$, such that $\dot{X}(t) = v$. This yields to the relation
  \[ \delta \leq \frac{\langle -\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)v, v \rangle}{\| v \|^2} \]

  By the min-max theorem, the largest eigenvalue is the maximum of $\frac{\langle -\text{Hess}_X(-\frac{1}{2}d(X, x^*)^2)v, v \rangle}{\| v \|^2}$ and the smallest its minimum over all $v \in T_{X(t)}M$. Thus we recover the initial estimation for the largest and smallest eigenvalue of $H$.

B. Non-convex setting

**Proposition 5.** Algorithm [7] applied to an $L$-smooth objective function $f$ for $N$-many steps, produces iterates $(y_k)$, such that

\[
\min_{k=0, \ldots, N} \| \nabla f(y_k) \|^2 \leq \frac{2L(f(x_0) - f(x^*))}{N}.
\]

**Proof.** The proof is standard and similar to theorem 2 in [Nesterov et al., 2018]. We have

\[
f(x_{k+1}) \leq f(y_k) - \frac{1}{2L} \| \nabla f(y_k) \|^2 \leq f(x_k) - \frac{1}{2L} \| \nabla f(y_k) \|^2.
\]

The first inequality follows by $L$-smoothness of $f$ and the definition of $x_{k+1}$ as a gradient step from $y_k$, while the second one follows by the optimality conditions in the definition of $y_k$.

We sum this inequality for $k = 0, \ldots, N$ and get

\[
f(x_0) - f(x^*) \geq f(x_0) - f(x^{N+1}) \geq \frac{N}{2L} \min_{k=0, \ldots, N} \| \nabla f(y_k) \|^2.
\]

Rearranging this inequality, we obtain

\[
\min_{k=0, \ldots, N} \| \nabla f(y_k) \|^2 \leq \frac{2L(f(x_0) - f(x^*))}{N}.
\]
C. Main analysis

Lemma 2. Assume that our working domain $A$ has diameter bounded by $D$. If $f : A \rightarrow \mathbb{R}$ is a geodesically convex and $L$-smooth function, the iterates $\{x_k\}$ produced by Algorithm 1 obtain the following accelerated rate:

$$f(x_k) - f(x^*) \leq \frac{4\zeta L \psi_0(x^*)}{k^2} + \hat{\theta}$$

where $\hat{\theta}$ depends on the properties of the manifold $M$ and such that $\mathcal{E}_k(x^*) \geq -\hat{\theta}$.

Proof. As in Nesterov et al. [2018], the proof relies on an estimate sequence of functions, defined as

$$\psi_k(x) = \psi^*_k + \frac{1}{2} \| \log_{v_k}(x) \|^2,$$

where $\psi^*_k$ is the minimum of $\psi_k$ which is yet to be specified.

The proof consists in establishing the following two inequalities – for a suitable choice of $\psi^*_k$ – from which one can prove the desired final result:

- **C1** $A_k f(x_k) \leq \psi^*_k$ (see definition of $A_k$ in Algorithm 1)
- **C2** $\psi_{k+1}(x) \leq \psi_k(x) + a_{k+1}(f(y_k) + \langle \nabla f(y_k), \log_{v_k}(x) \rangle + \hat{\theta})$, at least for $x = x^*$.

Proof C2.

Consider

$$\psi^*_k = \psi^*_k + a_{k+1} f(y_k) - \frac{\zeta a^2_{k+1}}{2} \| \nabla f(y_k) \|^2,$$

where $\zeta = \begin{cases} \sqrt{-k_{\min} D \coth(\sqrt{k_{\min} D})}, & k_{\min} < 0 \\ 1, & k_{\min} \geq 0. \end{cases}$

Assuming that $\hat{\theta}$ is such that $\langle \nabla f(y_k), \log_{v_k}(x) - \Gamma^v_{v_k} \log_{v_k}(x) \rangle \geq -\hat{\theta}$, we have

$$\begin{align*}
\psi_k(x) + a_{k+1}(f(y_k) + \langle \nabla f(y_k), \log_{v_k}(x) \rangle) \\
= \psi^*_k + \frac{1}{2} \| \log_{v_k}(x) \|^2 + a_{k+1}(f(y_k) + \langle \nabla f(y_k), \log_{v_k}(x) \rangle) \\
\geq \psi^*_k + a_{k+1} f(y_k) + \frac{1}{2} \| \log_{v_k}(x) \|^2 + a_{k+1}(\nabla f(y_k), \Gamma^v_{v_k} \log_{v_k}(x)) - a_{k+1} \hat{\theta} \\
= \psi^*_k + a_{k+1} f(y_k) + \frac{1}{2} \| \log_{v_k}(x) \|^2 + a_{k+1}(\Gamma^v_{v_k} \nabla f(y_k), \log_{v_k}(x)) - a_{k+1} \hat{\theta} \\
\geq \psi^*_k + a_{k+1} f(y_k) + \frac{1}{2} \| \log_{v_{k+1}}(x) \|^2 - \frac{\zeta a^2_{k+1}}{2} \| \nabla f(y_k) \|^2 - a_{k+1} \hat{\theta} \\
= \psi^*_{k+1} + \frac{1}{2} \| \log_{v_{k+1}}(x) \|^2 - a_{k+1} \hat{\theta} \\
= \psi_{k+1}(x) - a_{k+1} \hat{\theta},
\end{align*}$$

which concludes the proof of C2.

The last inequality follows from the definition of $v_{k+1}$ and using a trigonometric distance bound. First, we set $v_{k+1} = \exp_{v_k}(-a_{k+1} \Gamma^v_{v_k} \nabla f(y_k))$ and we get

$$\log_{v_k}(v_{k+1}) = -a_{k+1} \Gamma^v_{v_k} \nabla f(y_k).$$
Thus we have
\[
\frac{1}{2} \| \log_{v_k}(x) \|^2 + a_{k+1} (\Gamma_{v_k}^\alpha f(y_k), \log_{v_k}(x)) = \frac{1}{2} \| \log_{v_k}(x) \|^2 - (\log_{v_k}(v_{k+1}), \log_{v_k}(x)) \\
\geq \frac{1}{2} \| \log_{v_k}(x) \|^2 - \frac{\zeta}{2} \| \log_{v_k}(v_{k+1}) \|^2 = \frac{1}{2} \| \log_{v_k}(x) \|^2 - \frac{\zeta}{2} a_{k+1}^2 \| \Gamma_{v_k}^\alpha f(y_k) \|^2 \\
= \frac{1}{2} \| \log_{v_k}(x) \|^2 - \frac{\zeta a_{k+1}^2}{2} \| \log_{v_k}(x) \|^2.
\]
by the basic trigonometric distance bound (lemma 5 in (Zhang & Sra, 2016)) in the geodesic triangle $\Delta v_k v_{k+1} x$.

**Proof C1** We prove C1 by induction.

We assume that $A_k f(x_k) \leq \psi_k^*$ and we wish to prove that $A_{k+1} f(x_{k+1}) \leq \psi_{k+1}^*$.

\[
\psi_{k+1}^* = \psi_k^* + a_{k+1} f(y_k) - \frac{\zeta a_{k+1}^2}{2} \| \log_{v_k}(x) \|^2 \geq A_k f(x_k) + a_{k+1} f(y_k) - \frac{A_{k+1}}{2L} \| \log_{v_k}(x) \|^2 \geq A_{k+1} f(y_k) - \frac{A_{k+1}}{2L} \| \log_{v_k}(x) \|^2 = A_{k+1} f(y_k) - \frac{1}{2L} \| \log_{v_k}(x) \|^2, \]

where the last inequality follows from the definition of $x_{k+1}$ as a gradient step and $L$-smoothness of $f$.

**Combining C1 and C2** Now that we have established that both C1 and C2 hold, we get

\[
A_k f(x_k) \leq \psi_k^* \leq \psi_k(x^*) \leq \sum_{i=0}^{k-1} a_{i+1} (f(y_i) + (\log_{y_i}(x^*) + \tilde{\theta}) + \psi_0(x^*) \leq \sum_{i=0}^{k-1} a_{i+1} f(x^*) + \psi_0(x^*) + A_k \tilde{\theta} = A_k f(x^*) + \psi_0(x^*) + A_k \tilde{\theta},
\]

where the last inequality uses the geodesic-convexity property of the function $f$.

We can estimate a lower bound for $A_k$ from the equation $\frac{\zeta a_{k+1}^2}{A_k + a_{k+1}} = \frac{1}{L}$ (similarly to (Nesterov et al., 2018)). Namely $A_k \geq \frac{k^2}{4L}$ and we get an accelerated rate of convergence up to some error $\tilde{\theta}$, i.e.

\[
f(x_k) - f(x^*) \leq \frac{4 \zeta L \psi_0(x^*)}{k^2} + \tilde{\theta}.
\]

Using the fact that $\psi_0(x^*) = \frac{1}{2} \| \log_{v_0}(x^*) \|^2$, we get:

\[
f(x_k) - f(x^*) \leq \frac{2 \zeta L d(x_0, x^*)^2}{k^2} + \tilde{\theta} \leq \frac{2 \zeta L D^2}{k^2} + \tilde{\theta}.
\]

**D. The weakly-quasi-convex case**

We now turn our attention to the more general class of $\alpha$-weakly-quasi-convex functions. This requires a slight modification to Algorithm 1 by applying a restarting technique detailed in Algorithm 2.

The constant $c$ in the algorithm is chosen to be bigger than 1 ($c = 2$ in (Nesterov et al., 2018)).
Lemma 6. Algorithm 1 applied to an $\alpha$-weakly-convex function $f$ produces iterates $x_k$ satisfying

$$A_k(f(x_k) - f(x^*)) \leq (1 - \alpha)A_k(f(x_0) - f(x^*)) + \psi_0(x^*) + A_k\tilde{\theta}$$

Proof. We note that both $C1$ and $C2$ proven in Lemma 2 did not require convexity and we can therefore apply both inequalities to obtain:

$$A_k f(x_k) \leq \psi_k^* \leq \sum_{i=0}^{k-1} a_{i+1}((f(y_i) + \langle \text{grad} f(y_i), \log_{y_i}(x^*) \rangle + \tilde{\theta}) + \psi_0(x^*)$$

$$\leq \sum_{i=0}^{k-1} a_{i+1}((1 - \alpha)f(y_i) + \alpha f(x^*) + \tilde{\theta}) + \psi_0(x^*)$$

$$\leq \sum_{i=0}^{k-1} a_{i+1}((1 - \alpha)f(x_0) + \alpha f(x^*) + \tilde{\theta}) + \psi_0(x^*)$$

$$= A_k((1 - \alpha)f(x_0) + A_k\alpha f(x^*) + A_k\tilde{\theta} + \psi_0(x^*),$$

where the third inequality uses the fact that the function $f$ is $\alpha$-weakly-quasi-convex.

Thus

$$A_k(f(x_k) - f(x^*)) \leq A_k(1 - \alpha)(f(x_0) - f(x^*)) + \psi_0(x^*) + A_k\tilde{\theta}$$

\[\square\]

Theorem 4. Algorithm 2 applied to an $\alpha$-weakly-quasi-convex function produces a sequence of iterates $\{x_k\}_{k=1}^N$, such that

$$f(x_N) - f(x^*) \leq O\left(\frac{\zeta LD^2}{\alpha^3 N^2}\right) + \frac{c}{(c-1)\alpha} \tilde{\theta},$$

where $\tilde{\theta}$ is the same error as in the convex case and $c > 1$.

Proof. We first consider the first outer loop of Algorithm 2 for $i = 0$. Let $\epsilon_0 = f(x_0^0) - f(x^*)$. By Lemma 6 and the lower bound $A_k \geq \frac{k^2}{\alpha^2}$ established previously, we have that

$$f(x_k^0) - f(x^*) \leq (1 - \alpha)\epsilon_0 + \frac{22 LD^2}{k^2} + \tilde{\theta}.$$ 

We want to show that the LHS is less or equal than $(1 - \frac{\alpha}{c})\epsilon_0$, therefore it suffices that

$$(1 - \alpha)\epsilon_0 + \frac{22 LD^2}{k^2} + \tilde{\theta} \leq \left(1 - \frac{\alpha}{c}\right)\epsilon_0.$$ 

This implies that the algorithm is first restarted after at most $N_0 = \left\lfloor \frac{22 LD^2}{(c-1)c\epsilon_0 - \tilde{\theta}} \right\rfloor$ iterations.

Similarly between the $i^{th}$ and the $(i+1)^{th}$ restart we have that

$$f(x_i^k) - f(x^*) \leq (1 - \alpha)\left(1 - \frac{\alpha}{c}\right)^i \epsilon_0 + \frac{22 LD^2}{k^2} + \tilde{\theta} \leq \left(1 - \frac{\alpha}{c}\right)^{i+1} \epsilon_0,$$

which is equivalent to

$$\frac{22 LD^2}{k^2} \leq \frac{(c-1)\alpha}{c} \left(1 - \frac{\alpha}{c}\right)^i \epsilon_0 - \tilde{\theta},$$

or,

$$k \geq \frac{22 LD^2}{(c-1)\alpha \left(1 - \frac{\alpha}{c}\right)^i \epsilon_0 - \tilde{\theta}}.$$
Thus, between the \( i \)th and the \((i + 1)\)th restart we have at most

\[
N_i = \left\lfloor \sqrt{\frac{2\zeta LD^2}{(c-1)\alpha (1 - \frac{\alpha}{c})^i \epsilon_0 - \tilde{\theta}}} \right\rfloor
\]

steps (\( N_i \)-many steps suffice for the restart to happen).

Let \( d = \log_{1 - \frac{\alpha}{c}} \frac{\epsilon + M}{\epsilon_0} \). Then we obtain an \((\epsilon + M)\)-solution using algorithm 2 after \( d \)-many restarts. Define \( M = \frac{c}{(c-1)\alpha} \tilde{\theta} \).

Then, for any \( i = 0, \ldots, d \),

\[
\epsilon + M = \left(1 - \frac{\alpha}{c}\right)^d \epsilon_0 \\
\Rightarrow \frac{(c-1)\alpha}{c} \left(1 - \frac{\alpha}{c}\right)^d \epsilon_0 - \tilde{\theta} = \frac{(c-1)\alpha}{c} \epsilon \\
\Rightarrow \frac{(c-1)\alpha}{c} \left(1 - \frac{\alpha}{c}\right)^d \epsilon_0 - \left(1 - \frac{\alpha}{c}\right)^{d-i} \tilde{\theta} \geq \frac{(c-1)\alpha}{c} \epsilon,
\]

and finally

\[
\frac{(c-1)\alpha}{c} \left(1 - \frac{\alpha}{c}\right)^i \epsilon_0 - \tilde{\theta} \geq \frac{(c-1)\alpha}{c} \epsilon\left(1 - \frac{\alpha}{c}\right)^{d-i} \epsilon
\]

The third inequality holds because \((1 - \frac{\alpha}{c})^{d-i} \leq 1\) for any \( i = 0, \ldots, d \).

If algorithm 2 runs for \( N \)-many steps overall, we have

\[
N \leq \sum_{i=0}^{d} \left\lfloor \sqrt{\frac{2\zeta LD^2}{(c-1)\alpha (1 - \frac{\alpha}{c})^i \epsilon_0 - \tilde{\theta}}} \right\rfloor \\
\leq d + 1 + \sum_{i=0}^{d} \sqrt{\frac{2\zeta LD^2}{(c-1)\alpha \epsilon}} \left(1 - \frac{\alpha}{c}\right)^{d-i} \\
= d + 1 + \sqrt{\frac{2\zeta LD^2}{(c-1)\alpha \epsilon}} \sum_{i=0}^{d} \left(1 - \frac{\alpha}{c}\right)^{d-i} \\
= O\left(\sqrt{\frac{\zeta LD^2}{\alpha^3 \epsilon}}\right)
\]

similarly to the sequence of relations at the end of Theorem 4 in \cite{nesterov2018}. The last equality holds because the quantity \( \sum_{i=0}^{d} \left(1 - \frac{\alpha}{c}\right)^{d-i} \) is bounded by a constant depending only on \( \alpha \) and \( c \).

Indeed

\[
\sum_{i=0}^{d} \left(1 - \frac{\alpha}{c}\right)^{d-i} \leq \sum_{i=-\infty}^{d} \left(1 - \frac{\alpha}{c}\right)^{d-i} = \sum_{i=0}^{\infty} \left(1 - \frac{\alpha}{c}\right)^{2i} = \frac{1}{1 - \sqrt{1 - \frac{\alpha}{c}}} = \frac{1 + \sqrt{1 - \frac{\alpha}{c}}}{\frac{\alpha}{c}}
\]

We conclude that

\[
f(x_N) - f(x^*) \leq \epsilon + M \leq O\left(\frac{\zeta LD^2}{\alpha^3 N^2}\right) + \frac{c}{(c-1)\alpha} \tilde{\theta}.
\]