

GRADIENT METHOD FOR OPTIMIZATION ON RIEMANNIAN MANIFOLDS WITH LOWER BOUNDED CURVATURE*

O. P. FERREIRA[†], M. S. LOUZEIRO[†], AND L. F. PRUDENTE[†]

Abstract. The gradient method for minimizing a differentiable convex function on Riemannian manifolds with lower bounded sectional curvature is analyzed in this paper. An analysis of the method with three different finite procedures for determining the step size (namely, Lipschitz step size, adaptive step size, and Armijo's step size) is presented. The first procedure requires that the objective function has Lipschitz continuous gradient, which is not necessary for the other approaches. Convergence of the whole sequence to a minimizer, without any level set boundedness assumption, is proved. The iteration-complexity bound for functions with Lipschitz continuous gradient is also presented. Numerical experiments are provided to illustrate the effectiveness of the method in this new setting and certify the theoretical results. In particular, we consider the problem of finding the Riemannian center of mass and the so-called Karcher mean. Our numerical experiences indicate that the adaptive step size is a promising scheme that is worth considering.

Key words. gradient method, convex programming, Riemannian manifold, lower bounded curvature, iteration-complexity bound

AMS subject classifications. 90C30, 90C26, 49M37

DOI. 10.1137/18M1180633

1. Introduction. We consider the gradient method to solve the optimization problem defined by

$$(1.1) \quad \min\{f(p) : p \in \mathcal{M}\},$$

where the constraint set \mathcal{M} is endowed with the structure of a *complete Riemannian manifold with lower bounded curvature* and $f: \mathcal{M} \rightarrow \mathbb{R}$ is a *continuously differentiable convex function*. It is well known that, in several cases, by endowing \mathcal{M} with a suitable Riemannian metric, a Euclidian nonconvex constrained problem can be seen as a Riemannian convex unconstrained problem. In addition to this property, we will present some examples showing that, by endowing the set of constraints with a suitable Riemannian metric, the objective function can also become a *Riemannian Lipschitz gradient*. Consequently, the geometric and algebraic structures that come from the Riemannian metric make it possible to greatly reduce the computational cost for solving such problems. Indeed, it is also widely known that, in several contexts, the iteration complexity of the gradient method for convex optimization problems with Lipschitz gradient is much lower than for general nonconvex problems; see, for example, [6, 19, 30, 35, 41] and references therein. Furthermore, many Euclidian optimization problems are naturally posed in the Riemannian context; see [16, 19, 34, 35]. Then, to take advantage of the Riemannian geometric structure, it is preferable to treat these problems as those of finding singularities of gradient vector fields on Riemannian manifolds rather than using Lagrange multipliers or projection methods; see [25, 34, 36]. Accordingly, constrained optimization problems can be

*Received by the editors April 13, 2018; accepted for publication (in revised form) May 9, 2019; published electronically October 11, 2019.

<https://doi.org/10.1137/18M1180633>

Funding: This work was funded by the CNPQ (grants 302473/2017-3 and 408151/2016-1) and FAPEG (grant PRONEM-201710267000532).

[†]IME/UFG, Avenida Esperança, s/n, Campus Samambaia, CEP 74690-900, Goiânia, GO, Brazil (orizon@ufg.br, mauriciosilvalouzeiro@gmail.com, lfprudente@ufg.br).

viewed as unconstrained ones from a Riemannian geometry point of view. Moreover, Riemannian structures can also open up new research directions that aid in developing competitive algorithms; see [1, 16, 19, 29, 34, 35]. For this purpose, the concepts and techniques of optimization from Euclidian space have frequently been extended to Riemannian context in recent years. Papers dealing with this subject include [23, 24, 37, 38, 26, 41, 42].

The gradient method is one of the oldest methods for the minimization of a differentiable function in Euclidian space. Despite it having a slow convergence rate, the simplicity of implementation, the low memory requirements, and low cost per iteration make the gradient method quite attractive for solving large-scale optimization problems. Indeed, the computational cost per iteration is mildly dependent on the dimension of the problem, yielding computational efficiency for this method; see [19, 28, 31]. In addition, the gradient method is the starting point for designing many more sophisticated and efficient algorithms, including the fast gradient method, accelerated gradient method, and Barzilai–Borwein method; see [27, 40] for a comprehensive study on this subject. To the best of our knowledge, the gradient method was the first optimization method to be considered in a Riemannian setting.

In order to deal with constrained optimization problems in the Euclidian space, Luenberger [25] proposed and established important convergence properties of the gradient method by using the Riemannian structure of the constraint set induced by the Euclidian structure. Since then, the gradient method has been studied in the general Riemannian manifold. Some early works dealing with this method include [18, 36, 34, 30]. However, the convergence results obtained in these works demand that the initial points of the sequence belong to a bounded level set of the objective function, establishing only that all its cluster points are stationary. By assuming convexity of the objective function and that the manifold has *nonnegative curvature*, it has been proven in [11] that, for a suitable choice of step size and without any level set boundedness assumption, the whole sequence converges to a solution. Recently, new important properties of the gradient method have been obtained in Riemannian settings. For instance, Zhang and Sra [42] provided iteration-complexity bounds for convex optimization problems on Hadamard manifolds. In [8], Boumal, Absil, and Cartis established iteration-complexity bounds without any assumption on the convexity of the problem and curvature of the manifold. In [7] the gradient method is considered to compute the Karcher mean, which is a strong convex function in the cone of symmetric positive definite matrices endowed with a suitable Riemannian metric. In [2], the properties of the gradient method for the problem of finding the global Riemannian center of mass of a set of data points on a Riemannian manifold are studied. In [5], the convergence analysis of the gradient method is extended to the Hadamard setting for continuously differentiable functions that satisfy the Kurdyka–Łojasiewicz inequality.

By the above we see that the gradient method remains a subject of considerable interest. In spite of its long history, the full convergence of the sequence generated by the gradient method in a general Riemannian manifold has not yet been established. However, as far as we know, the full convergence of the sequence generated by the gradient method under convexity of the objective function and *lower boundedness of the curvature* of the Riemannian manifold is a new contribution of this paper, which adds important results on the convergence theory available by this method. Our analysis of the method is presented with three different finite procedures to determine the step size, namely, Lipschitz step size, adaptive step size, and Armijo's step size. Note that we use a recent inequality established in [37, 38]. Numerical

experiments illustrate the effectiveness of the method in this new setting and certify the theoretical results obtained. In particular, we consider the problem of finding the Riemannian center of mass and the so-called Karcher mean. Our experiments indicate that adaptive step size is a promising scheme that is worth further consideration.

This paper is organized as follows. Section 2 presents some definitions and preliminary results related to the Riemannian geometry that are important throughout our study. In section 3, we state the gradient algorithm and the three different finite procedures for determining the step size. Section 3.1 is devoted to the asymptotic convergence analysis of the method, and in section 3.2 the iteration-complexity bound is presented. Section 4 provides some examples of functions satisfying the assumptions of our results in the previous sections. In section 5, we present some numerical experiments to illustrate the behavior of the method. The last section contains some conclusions.

2. Notation and basic concepts. In this section, we recall some concepts, notation, and basic results about Riemannian manifolds. For more details we refer the reader to [13, 33, 36, 30].

We denote by $T_p\mathcal{M}$ the *tangent space* of a finite-dimensional Riemannian manifold \mathcal{M} at p . The corresponding norm associated to the Riemannian metric $\langle \cdot, \cdot \rangle$ is denoted by $\| \cdot \|$. We use $\ell(\alpha)$ to denote the length of a piecewise smooth curve $\alpha: [a, b] \rightarrow \mathcal{M}$. The Riemannian distance between p and q in \mathcal{M} is denoted by $d(p, q)$, which induces the original topology on \mathcal{M} , namely, (\mathcal{M}, d) is a complete metric space. Denote by $\mathcal{X}(\mathcal{M})$ the space of smooth vector fields on \mathcal{M} . Let ∇ be the Levi-Civita connection associated to $(\mathcal{M}, \langle \cdot, \cdot \rangle)$. For each $t \in [a, b]$ and a piecewise smooth curve $\alpha: [a, b] \rightarrow \mathcal{M}$, ∇ induces an isometry relative to $\langle \cdot, \cdot \rangle$, $P_{\alpha, a, t}: T_{\alpha(a)}\mathcal{M} \rightarrow T_{\alpha(t)}\mathcal{M}$ defined by $P_{\alpha, a, t} v = V(t)$, where V is the unique vector field on the curve α such that $\nabla_{\alpha'(t)} V(t) = 0$ and $V(a) = v$. The isometry $P_{\alpha, a, t}$ is called *parallel transport* along α , joining $\alpha(a)$ to $\alpha(t)$, and when there is no confusion it will be denoted by $P_{\alpha, p, q}$. A vector field V along a smooth curve γ is said to be *parallel* iff $\nabla_{\gamma'} V = 0$. If γ' itself is parallel, we say that γ is a *geodesic*. The restriction of a geodesic to a closed bounded interval is called a *geodesic segment*.

A geodesic segment joining p to q in \mathcal{M} is said to be *minimal* if its length is equal to $d(p, q)$. A Riemannian manifold is *complete* if the geodesics are defined for any values of $t \in \mathbb{R}$. The Hopf–Rinow theorem asserts that any pair of points in a complete Riemannian manifold \mathcal{M} can be joined by a (not necessarily unique) minimal geodesic segment. Owing to the completeness of the Riemannian manifold \mathcal{M} , for each $p \in \mathcal{M}$ the *exponential map* $\exp_p: T_p\mathcal{M} \rightarrow \mathcal{M}$ is given by $\exp_p v = \gamma(1)$, where $\gamma(0) = p$ and $\gamma'(0) = v$. In this paper, all manifolds are assumed to be connected, finite dimensional, and complete. For $f: \mathcal{D} \rightarrow \mathbb{R}$ a differentiable function on the open set $\mathcal{D} \subset \mathcal{M}$, the Riemannian metric induces the mapping $f \mapsto \text{grad } f$, which associates its *gradient* via the following rule: $\langle \text{grad } f(p), X(p) \rangle := df(p)X(p)$ for all $p \in \mathcal{D}$. For a twice-differentiable function, the mapping $f \mapsto \text{hess } f$ associates its *Hessian* via the rule $\langle \text{hess } f X, X \rangle := d^2 f(X, X)$ for all $X \in \mathcal{X}(\mathcal{D})$, where the latter equalities imply that $\text{hess } f X = \nabla_X \text{grad } f$ for all $X \in \mathcal{X}(\mathcal{D})$. We recall some concepts and basic properties about convexity in the Riemannian context. For more details see, for example, [36, 30, 37].

For any two points $p, q \in \mathcal{M}$, Γ_{pq} denotes the set of all geodesic segments $\gamma: [0, 1] \rightarrow \mathcal{M}$ with $\gamma(0) = p$ and $\gamma(1) = q$. We use Γ_{pq}^Ω to denote the set of all $\gamma \in \Gamma_{pq}$ such that $\gamma(t) \in \Omega$ for all $t \in [0, 1]$. A nonempty subset $\Omega \subset \mathcal{M}$ is said to be *weakly convex* if, for any $p, q \in \Omega$, there is a minimal geodesic segment joining p to q .

that belongs to Ω . A function $f: \mathcal{D} \rightarrow \mathbb{R}$ is said to be *convex* on the set $\Omega \subset \mathcal{D}$ if Ω is weakly convex and, for any $p, q \in \Omega$ and $\gamma \in \Gamma_{pq}^\Omega$, the composition $f \circ \gamma: [0, 1] \rightarrow \mathbb{R}$ is a convex function on $[0, 1]$, i.e., $(f \circ \gamma)(t) \leq (1-t)f(p) + tf(q)$ for all $t \in [0, 1]$; see [37].

The lemma below plays an important role in the following sections; its proof, with some minor technical adjustments, can be found in [37, Lemma 3.2]; see also [38]. To simplify the notation, let

$$(2.1) \quad \kappa < 0, \quad \hat{\kappa} := \sqrt{|\kappa|}.$$

LEMMA 2.1. *Let \mathcal{M} be a Riemannian manifold with sectional curvature $K \geq \kappa$, and let $\hat{\kappa}$ be as defined in (2.1). Assume that f is differentiable and convex on the set $\Omega \subset \mathcal{M}$, $p \in \Omega$, and $\gamma: [0, \infty) \rightarrow \mathcal{M}$ is defined by $\gamma(t) = \exp_p(-t \operatorname{grad} f(p))$. Then, for any $t \in [0, \infty)$ and $q \in \Omega$, it holds that*

$$\begin{aligned} \cosh(\hat{\kappa}d(\gamma(t), q)) &\leq \cosh(\hat{\kappa}d(p, q)) + \hat{\kappa} \cosh(\hat{\kappa}d(p, q)) \sinh(\hat{\kappa}t\|\operatorname{grad} f(p)\|) \\ &\quad \times \left[\frac{t\|\operatorname{grad} f(p)\|}{2} - \frac{\tanh(\hat{\kappa}d(p, q))}{\hat{\kappa}d(p, q)} \frac{f(p) - f(q)}{\|\operatorname{grad} f(p)\|} \right], \end{aligned}$$

and consequently, the following inequality holds:

$$\begin{aligned} d^2(\gamma(t), q) &\leq d^2(p, q) + \frac{\sinh(\hat{\kappa}t\|\operatorname{grad} f(p)\|)}{\hat{\kappa}} \\ &\quad \times \left[t\|\operatorname{grad} f(p)\| \frac{\hat{\kappa}d(p, q)}{\tanh(\hat{\kappa}d(p, q))} - \frac{2}{\|\operatorname{grad} f(p)\|} (f(p) - f(q)) \right]. \end{aligned}$$

Next we present the definition of the Lipschitz continuous gradient vector field; see [12].

DEFINITION 2.2. *Let f be a differentiable function on the set \mathcal{D} . The gradient vector field of f is said to be Lipschitz continuous on \mathcal{D} with constant $L \geq 0$ if, for any $p, q \in \mathcal{D}$ and $\gamma \in \Gamma_{pq}^\mathcal{D}$, it holds that $\|P_{\gamma,p,q} \operatorname{grad} f(p) - \operatorname{grad} f(q)\| \leq L\ell(\gamma)$.*

The norm of the Hessian $\operatorname{hess} f$ at $p \in \mathcal{M}$ is given by

$$(2.2) \quad \|\operatorname{hess} f(p)\| := \sup\{\|\operatorname{hess} f(p)v\| : v \in T_p\mathcal{M}, \|v\| = 1\}.$$

In the following result we present a characterization for twice continuously differentiable functions with Lipschitz continuous gradient vector field; the proof of its Euclidian counterpart is similar and thus omitted.

LEMMA 2.3. *Let $f: \mathcal{D} \rightarrow \mathbb{R}$ be a twice continuously differentiable function. The gradient vector field of f is Lipschitz continuous with constant $L \geq 0$ if and only if there exists $L \geq 0$ such that $\|\operatorname{hess} f(p)\| \leq L$ for all $p \in \mathcal{D}$.*

The next lemma can be found in [6, Corollary 2.1] with minor adjustments. Its proof follows from the fundamental theorem of calculus.

LEMMA 2.4. *Let f be a differentiable function on the set \mathcal{D} and let $a > 0$. Assume that $\operatorname{grad} f$ is Lipschitz continuous on \mathcal{D} with constant $L \geq 0$ and $p \in \mathcal{D}$. If $\exp_p(-t \operatorname{grad} f(p)) \in \mathcal{D}$ for all $t \in [0, a]$, then it holds that*

$$f(\exp_p(-t \operatorname{grad} f(p))) \leq f(p) - \left(1 - \frac{L}{2}t\right)t\|\operatorname{grad} f(p)\|^2 \quad \text{for all } t \in [0, a].$$

Note that if $\mathcal{D} = \mathcal{M}$, then the condition $\exp_p(-t \operatorname{grad} f(p)) \in \mathcal{D}$ for all $t \in [0, a]$ in Lemma 2.4 plays no role. In the following example we present a function satisfying all the assumptions of Lemma 2.4 for the case when $\mathcal{D} \neq \mathcal{M}$.

Example 2.5. Let $\mathcal{M} = \{p \in \mathbb{R}^n : \|p\| = 1\}$ be the Euclidian sphere and let $q \in \mathcal{M}$. Define $\varphi_q(p) := d^2(p, q)/2$ for all $p \in \mathcal{M}$. The function φ_q is differentiable in $\mathcal{D} := \{p \in \mathcal{M} : d(p, q) < 5\pi/6\}$ and convex in $\Omega := \{p \in \mathcal{M} : d(p, q) \leq \pi/2\}$. Furthermore, $\operatorname{grad} \varphi_q$ is Lipschitz continuous on \mathcal{D} because $\operatorname{hess} \varphi_q$ is continuous in $\mathcal{M} \setminus \{-q\} \supset \mathcal{D}$. Indeed, combining [17, Lemma 3] with Lemma 2.3 we conclude that

$$L = \sup_{p \in \mathcal{D}} \frac{|\langle p, q \rangle \arccos \langle p, q \rangle|}{\sqrt{1 - \langle p, q \rangle^2}} = \frac{5\pi}{6} \sqrt{3}.$$

Since $\operatorname{grad} \varphi_q(p) = -\exp_p^{-1} q$ for all $p \in \mathcal{M} \setminus \{-q\}$, after some calculations we conclude that $d(\exp_p(-t \operatorname{grad} \varphi_q(p)), p) \leq td(p, q)$ for all $p \in \mathcal{D}$. Hence, letting $p \in \Omega$ we have

$$d(\exp_p(-t \operatorname{grad} \varphi_q(p)), q) \leq d(\exp_p(-t \operatorname{grad} \varphi_q(p)), p) + d(p, q) \leq (t+1) \frac{\pi}{2},$$

and then $\exp_p(-t \operatorname{grad} \varphi_q(p)) \in \mathcal{D}$ for all $t \in [0, 1/L]$. For more details about the function φ_q see [17].

The following concept will be useful in the analysis of the sequence generated by the gradient method. In fact, as we shall prove, the sequence generated by this method satisfies the following definition.

DEFINITION 2.6. A sequence $\{y_k\}$ in the complete metric space (\mathcal{M}, d) is *quasi-Fejér convergent* to a set $W \subset \mathcal{M}$ if, for every $w \in W$, there exists a sequence $\{\epsilon_k\} \subset \mathbb{R}$ such that $\epsilon_k \geq 0$, $\sum_{k=1}^{\infty} \epsilon_k < +\infty$, and $d^2(y_{k+1}, w) \leq d^2(y_k, w) + \epsilon_k$ for all $k = 0, 1, \dots$.

The main property of a quasi-Fejér sequence is stated in the next result, and its proof is similar that in [9] with the Euclidian distance replaced by the Riemannian one.

THEOREM 2.7. Let $\{y_k\}$ be a sequence in the complete metric space (\mathcal{M}, d) . If $\{y_k\}$ is quasi-Fejér convergent to a nonempty set $W \subset \mathcal{M}$, then $\{y_k\}$ is bounded. Furthermore, if a cluster point \bar{y} of $\{y_k\}$ belongs to W , then $\lim_{k \rightarrow \infty} y_k = \bar{y}$.

The study of the gradient method for convex functions is well understood for the Riemannian manifold with nonnegative sectional curvature and Hadamard manifolds; see [12, 41, 42]. In order to increase the domain of applications of the method, we hereafter assume that \mathcal{M} is a complete Riemannian manifold with sectional curvature $K \geq \kappa$, where $\kappa < 0$, unless the contrary is explicitly stated.

3. The Riemannian gradient method. In this section we state the Riemannian gradient method used to solve (1.1) and the strategies for choosing the step size that will be used in our analysis.

Let $f: \mathcal{D} \rightarrow \mathbb{R}$ be a continuously differentiable function, $\mathcal{D} \subset \mathcal{M}$ be an open set, Ω^* be the *solution set* of the problem (1.1), $f^* := \inf_{p \in \mathcal{D}} f(p)$ be the *optimum value* of f , and $c \in \mathbb{R}$. From now on, we assume that Ω^* is nonempty and f is convex on the sublevel set $\mathcal{L}_c f$, where

$$\mathcal{L}_c f := \{p \in \mathcal{M} : f(p) \leq c\} \subset \mathcal{D}.$$

The *Riemannian gradient algorithm* to solve (1.1) is stated as Algorithm 3.1.

Algorithm 3.1. Gradient algorithm for a Riemannian manifold \mathcal{M} .

Step 0. Let $p_0 \in \mathcal{L}_c f$. Set $k = 0$.

Step 1. If $\text{grad } f(p_k) = 0$, then **stop**; otherwise, choose a step size $t_k > 0$ and compute

$$(3.1) \quad p_{k+1} := \exp_{p_k}(-t_k \text{grad } f(p_k)).$$

Step 2. Set $k \rightarrow k + 1$ and proceed to **Step 1**.

In the following we present three different strategies for choosing the step size $t_k > 0$ in Algorithm 3.1. In the first strategy we assume that $\text{grad } f$ is Lipschitz continuous.

STRATEGY 3.2 (Lipschitz step size). *Assume that $\text{grad } f$ is Lipschitz continuous on \mathcal{D} with constant $L \geq 0$ and that $\exp_p(-t \text{grad } f(p)) \in \mathcal{D}$ for all $p \in \mathcal{L}_c f$ and $t \in [0, 1/L]$. Let $\varepsilon > 0$ and take*

$$(3.2) \quad \varepsilon < t_k \leq \frac{1}{L}.$$

Despite knowing that $\text{grad } f$ is Lipschitz continuous, in general, the Lipschitz constant is not computable. The next strategy can be used to compute the step size without any Lipschitz condition. However, as we shall show, if $\text{grad } f$ is Lipschitz with constant $L > 0$, then the step size computed is an approximation to the step size $1/L$; see [4].

STRATEGY 3.3 (adaptive step size). *Take $\beta \in (0, 1/2]$, $L_0 > 0$, and $\eta > 1$. Set $t_k := L_k^{-1}$, where $L_k := \eta^{i_k} L_{k-1}$ and*

$$(3.3) \quad i_k := \min\{i: f(\gamma_k(\tau_i)) \leq f(p_k) - \beta \tau_i \|\text{grad } f(p_k)\|^2, i = 0, 1, \dots\},$$

where $\tau_i := (\eta^i L_{k-1})^{-1}$ and $\gamma_k(\tau_i) := \exp_{p_k}(-\tau_i \text{grad } f(p_k))$.

STRATEGY 3.4 (Armijo's step size). *Choose $\beta \in (0, 1)$ and take*

$$(3.4) \quad t_k := \max\{2^{-i}: f(\gamma_k(2^{-i})) \leq f(p_k) - \beta 2^{-i} \|\text{grad } f(p_k)\|^2, i = 0, 1, \dots\},$$

where $\gamma_k(2^{-i}) := \exp_{p_k}(-2^{-i} \text{grad } f(p_k))$.

Remark 3.5. If $\mathcal{D} = \mathcal{M}$, then the condition $\exp_p(-t \text{grad } f(p)) \in \mathcal{D}$ for all $t \in [0, a]$ in Strategy 3.2 plays no role. Recall that the function in Example 2.5 satisfies this condition for $\mathcal{D} \neq \mathcal{M}$.

Remark 3.6. Strategy 3.3 can be seen as an Armijo-type line search where the first trial step size at iteration k is set equal to t_{k-1} . Indeed, taking $L_0 = 1$ and $\eta = 2$, the inequality in (3.3) can be equivalently rewritten as

$$f(\gamma_k(2^{-i} t_{k-1})) \leq f(p_k) - \beta 2^{-i} t_{k-1} \|\text{grad } f(p_k)\|^2.$$

The proof of the well-definedness of Strategies 3.3 and 3.4 follows the usual arguments and will be omitted. Hence, the sequence $\{p_k\}$ generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4 is well defined. Finally, we remark that, due to f being convex, $\text{grad } f(p) = 0$ if and only if $p \in \Omega^*$. Therefore, *from now on we assume that $\text{grad } f(p_k) \neq 0$, or equivalently, $p_k \notin \Omega^*$ for all $k = 0, 1, \dots$*

3.1. Asymptotic convergence analysis. In this section our goal is to prove that the sequence $\{p_k\}$, generated by the gradient method with Strategies 3.2, 3.3, or 3.4, converges to a solution of problem (1.1).

LEMMA 3.7. *Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4. Then,*

$$(3.5) \quad f(p_{k+1}) \leq f(p_k) - \nu t_k \|\text{grad } f(p_k)\|^2, \quad k = 0, 1, \dots,$$

where $\nu = 1/2$ for Strategy 3.2, and $\nu = \beta$ for Strategies 3.3 and 3.4. Consequently, $\{f(p_k)\}$ is a nonincreasing sequence and $\lim_{k \rightarrow +\infty} t_k \|\text{grad } f(p_k)\|^2 = 0$.

Proof. For Strategies 3.3 and 3.4, inequality (3.5) follows directly from (3.3) and (3.4), respectively. Now, we assume that $\{p_k\}$ is generated by using Strategy 3.2. In this case, Lemma 2.4 implies that

$$f(p_{k+1}) = f(\exp_{p_k}(-t_k \text{grad } f(p_k))) \leq f(p_k) - \left(1 - \frac{L}{2} t_k\right) t_k \|\text{grad } f(p_k)\|^2$$

for all $k = 0, 1, \dots$. Hence, taking into account (3.2), we have $1/2 \leq (1 - Lt_k/2)$ and (3.5) follows. Therefore, (3.5) holds for $\{p_k\}$ generated by using one of the three strategies. It is immediate from (3.5) that $\{f(p_k)\}$ is nonincreasing. Moreover, (3.5) implies that

$$\sum_{k=0}^{\ell} t_k \|\text{grad } f(p_k)\|^2 \leq \frac{1}{\nu} \sum_{k=0}^{\ell} f(p_k) - f(p_{k+1}) \leq \frac{1}{\nu} (f(p_0) - f^*)$$

for each nonnegative integer ℓ , which implies that $t_k \|\text{grad } f(p_k)\|^2$ goes to zero as k goes to infinity, completing the proof. \square

Remark 3.8. Whenever $\text{grad } f$ is Lipschitz continuous on \mathcal{D} with constant $L \geq 0$, the step size in Strategy 3.3 can be seen as an approximation for $1/L$. Indeed, since $L_0 > 0$ and $\eta > 1$ in Strategy 3.3, we conclude that $t_k := L_k^{-1} \leq L_{k-1}^{-1} = t_{k-1}$ for all $k = 0, 1, \dots$. Thus, $t_k \leq 1/L_0$ for all $k = 0, 1, \dots$. If $L_0 \geq L$, then it follows from (3.5) that $t_k = 1/L_0$ for all $k = 0, 1, \dots$. Therefore, for Strategy 3.3 we assume $L_0 \leq L$. In this case, (3.5) holds for $t_k = 1/L$ and then (3.3) implies that $1/(\eta L) \leq t_k$. Hence,

$$(3.6) \quad \frac{1}{\eta L} \leq t_k \leq \frac{1}{L_0}, \quad k = 0, 1, \dots$$

Let $p_0 \in \mathcal{L}_c f$. By Lemma 3.7, we define the constant $\rho > 0$ as follows:

$$(3.7) \quad \sum_{k=0}^{\infty} t_k^2 \|\text{grad } f(p_k)\|^2 \leq \rho := \begin{cases} 2[f(p_0) - f^*]/L & \text{for Strategy 3.2,} \\ [f(p_0) - f^*]/(\beta L_0) & \text{for Strategy 3.3,} \\ [f(p_0) - f^*]/\beta & \text{for Strategy 3.4.} \end{cases}$$

In the following result in particular, we bound the sequence $\{p_k\}$ generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4.

LEMMA 3.9. *Let $q \in \Omega^*$ and let $\{p_k\}$ be the sequence generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4. Then, it holds that*

$$(3.8) \quad d(p_{k+1}, q) \leq \frac{1}{\hat{\kappa}} \cosh^{-1} \left(\cosh(\hat{\kappa} d(p_0, q)) e^{\frac{1}{2}(\hat{\kappa}\sqrt{\rho}) \sinh(\hat{\kappa}\sqrt{\rho})} \right), \quad k = 0, 1, \dots$$

Proof. Applying the first inequality of Lemma 2.1, with $t = t_k$ and $p = p_k$, we have $p_{k+1} = \gamma(t_k)$, and taking into account that $q \in \Omega^*$, we conclude that

$$\cosh(\hat{\kappa}d(p_{k+1}, q)) \leq \cosh(\hat{\kappa}d(p_k, q)) \left[1 + (\hat{\kappa}t_k \|\text{grad } f(p_k)\|)^2 \frac{\sinh(\hat{\kappa}t_k \|\text{grad } f(p_k)\|)}{2\hat{\kappa}t_k \|\text{grad } f(p_k)\|} \right]$$

for all $k = 0, 1, \dots$, where $\hat{\kappa}$ is defined in (2.1). Since (3.7) implies $t_k \|\text{grad } f(p_k)\| \leq \sqrt{\rho}$ for all $k = 0, 1, \dots$, and the map $(0, +\infty) \ni t \mapsto \sinh(t)/t$ is increasing, we conclude that

$$\cosh(\hat{\kappa}d(p_{k+1}, q)) \leq \cosh(\hat{\kappa}d(p_k, q)) [1 + a(t_k \|\text{grad } f(p_k)\|)^2], \quad k = 0, 1, \dots,$$

where $a := \hat{\kappa}(\sinh(\hat{\kappa}\sqrt{\rho}))/ (2\sqrt{\rho})$. Now note that the last inequality implies that

$$\cosh(\hat{\kappa}d(p_{k+1}, q)) \leq \cosh(\hat{\kappa}d(p_k, q)) e^{a(t_k \|\text{grad } f(p_k)\|)^2}, \quad k = 0, 1, \dots,$$

Therefore, by using (3.7), it follows that $\cosh(\hat{\kappa}d(p_{k+1}, q)) \leq \cosh(\hat{\kappa}d(p_0, q)) e^{a\rho}$, which is equivalent to (3.8) by considering the definition of $\hat{\kappa}$ in (2.1). \square

Let us define the following auxiliary constant:

$$(3.9) \quad C_{\rho, \kappa}^q := \frac{\sinh(\hat{\kappa}\sqrt{\rho})}{\hat{\kappa}\sqrt{\rho}} \left[1 + \cosh^{-1} \left(\cosh(\hat{\kappa}d(p_0, q)) e^{\frac{1}{2}(\hat{\kappa}\sqrt{\rho}) \sinh(\hat{\kappa}\sqrt{\rho})} \right) \right],$$

where ρ is defined in (3.7).

LEMMA 3.10. *Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4. Then, for each $q \in \Omega^*$, it holds that*

$$(3.10) \quad d^2(p_{k+1}, q) \leq d^2(p_k, q) + \frac{t_k}{\nu} C_{\rho, \kappa}^q [f(p_k) - f(p_{k+1})] + 2t_k [f^* - f(p_k)]$$

for all $k = 0, 1, \dots$, where $\nu = 1/2$ for Strategy 3.2 and $\nu = \beta$ for Strategies 3.3 and 3.4.

Proof. Define $\gamma_k(t) = \exp_{p_k}(-t \text{grad } f(p_k))$ for all $t \in [0, +\infty)$. Then, $\gamma_k(0) = p_k$ and, from (3.1), we obtain $\gamma_k(t_k) = p_{k+1}$. Applying the second inequality of Lemma 2.1 with $\gamma = \gamma_k$ we conclude, after some manipulations, that

$$(3.11) \quad d^2(p_{k+1}, q) \leq d^2(p_k, q) + \frac{\sinh(\hat{\kappa}t_k \|\text{grad } f(p_k)\|)}{\hat{\kappa}t_k \|\text{grad } f(p_k)\|} \left[t_k^2 \|\text{grad } f(p_k)\|^2 \frac{\hat{\kappa}d(p_k, q)}{\tanh(\hat{\kappa}d(p_k, q))} + 2t_k [f^* - f(p_k)] \right]$$

for all $k = 0, 1, \dots$. On the other hand, $t/\tanh(t) \leq 1+t$ for all $t \geq 0$, and the map $(0, +\infty) \ni t \mapsto \sinh(t)/t$ is increasing and bounded below by 1. Thus, taking into account that (3.7) implies $t_k \|\text{grad } f(p_k)\| \leq \sqrt{\rho}$ for all $k = 0, 1, \dots$, and considering $f^* - f(p_k) \leq 0$ for all $k = 0, 1, \dots$, we conclude from (3.11) that

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + \frac{\sinh(\hat{\kappa}\sqrt{\rho})}{\hat{\kappa}\sqrt{\rho}} t_k^2 \|\text{grad } f(p_k)\|^2 [1 + \hat{\kappa}d(p_k, q)] + 2t_k [f^* - f(p_k)]$$

for all $k = 0, 1, \dots$, where ρ is defined in (3.7). Thus, by Lemma 3.7, we obtain

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + \frac{t_k}{\nu} \frac{\sinh(\hat{\kappa}\sqrt{\rho})}{\hat{\kappa}\sqrt{\rho}} [1 + \hat{\kappa}d(p_k, q)] [f(p_k) - f(p_{k+1})] + 2t_k [f^* - f(p_k)]$$

for all $k = 0, 1, \dots$. Therefore, by Lemma 3.9 and (3.9), we have (3.10), which concludes the proof. \square

Finally, we are ready to prove the full convergence of $\{p_k\}$ to a minimizer of f .

THEOREM 3.11. *Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategies 3.2, 3.3, or 3.4. Then $\{p_k\}$ converges to a solution of the problem in (1.1).*

Proof. First note that (3.2), (3.4), and (3.6) imply $0 < t_k \leq 1/L$, $0 < t_k \leq 1$, or $0 < t_k \leq 1/L_0$ for all $k = 0, 1, \dots$, for Strategies 3.2, 3.4 or 3.3, respectively. Let $\Gamma := \max\{1/L, 1, 1/L_0\}$. Thus, for Strategies 3.2, 3.3, or 3.4 we conclude from Lemma 3.10 that

$$d^2(p_{k+1}, q) \leq d^2(p_k, q) + \frac{1}{\nu} \Gamma C_{\rho, \kappa}^q [f(p_k) - f(p_{k+1})], \quad k = 0, 1, \dots,$$

for all $q \in \Omega^*$. Considering that $\sum_{i=0}^{\infty} [f(p_k) - f(p_{k+1})] \leq [f(p_0) - f^*]$, we conclude that $\{p_k\}$ is quasi-Fejér convergent to Ω^* . Therefore, since Ω^* is nonempty, the sequence $\{p_k\}$ is bounded. Let \bar{p} be a cluster point of $\{p_k\}$ and $\{p_{k_j}\}$ be a subsequence of $\{p_k\}$ such that $\lim_{j \rightarrow \infty} p_{k_j} = \bar{p}$. Since the sequence $\{t_k\}$ has a cluster point $\bar{t} \in [0, \Gamma]$, it follows from Lemma 3.7 that $\lim_{k \rightarrow \infty} t_k \|\text{grad } f(p_k)\|^2 = 0$. We analyze the following two possibilities:

$$(a) \quad \bar{t} > 0, \quad (b) \quad \bar{t} = 0.$$

Assume that (a) holds. In this case, considering that $\lim_{k \rightarrow \infty} t_k \|\text{grad } f(p_k)\|^2 = 0$ and $\text{grad } f$ is continuous, we conclude that

$$0 = \lim_{j \rightarrow \infty} t_{k_j} \|\text{grad } f(p_{k_j})\| = \bar{t} \|\text{grad } f(\bar{p})\|.$$

Hence, $\text{grad } f(\bar{p}) = 0$ and then $\bar{p} \in \Omega^*$. Note that if Strategy 3.2 is used, then \bar{t} satisfies only (a). Now, we assume that (b) holds. In this case Strategy 3.3 or 3.4 is used. First, assume Algorithm 3.1 with Strategy 3.3. Since $\{t_{k_j}\}$ converges to $\bar{t} = 0$ and $\{t_k\}$ is nonincreasing, it follows that $\{t_k\}$ converges to $\bar{t} = 0$. Hence, taking $r \in \mathbb{N}$, we can conclude that $t_k < (\eta^r L_0)^{-1}$ for k sufficiently large. Thus, (3.3) implies

$$f(\exp_{p_k}((\eta^r L_0)^{-1}[-\text{grad } f(p_{k_j})])) > f(p_k) - (\eta^r L_0)^{-1} \beta \|\text{grad } f(p_k)\|^2.$$

Letting k go to $+\infty$ in the above inequality and taking into account that $\text{grad } f$ and the exponential mapping are continuous, we obtain

$$f(\exp_{\bar{p}}((\eta^r L_0)^{-1}[-\text{grad } f(\bar{p})])) \geq f(\bar{p}) - (\eta^r L_0)^{-1} \beta \|\text{grad } f(\bar{p})\|^2.$$

The latter inequality is equivalent to

$$-\frac{f(\exp_{\bar{p}}((\eta^r L_0)^{-1}[-\text{grad } f(\bar{p})])) - f(\bar{p})}{(\eta^r L_0)^{-1}} \leq \beta \|\text{grad } f(\bar{p})\|^2.$$

Thus, letting r go to $+\infty$, we obtain $\|\text{grad } f(\bar{p})\|^2 \leq \beta \|\text{grad } f(\bar{p})\|^2$, which implies $\text{grad } f(\bar{p}) = 0$, i.e., $\bar{p} \in \Omega^*$. Therefore, since $\{p_k\}$ is quasi-Fejér convergent to Ω^* , we conclude from Theorem 2.7 that $\{p_k\}$ converges to \bar{p} . Finally, assume that Strategy 3.4 is used. Since $\{t_{k_j}\}$ converges to $\bar{t} = 0$, taking $r \in \mathbb{N}$, we conclude that $t_{k_j} < 2^{-r}$ for j sufficiently large. Thus, Armijo's condition (3.4) is not satisfied for 2^{-r} , i.e.,

$$f(\exp_{p_{k_j}}(2^{-r}[-\text{grad } f(p_{k_j})])) > f(p_{k_j}) - 2^{-r} \beta \|\text{grad } f(p_{k_j})\|^2.$$

Letting j go to $+\infty$ in the above inequality and taking into account that $\text{grad } f$ and the exponential mapping are continuous, we obtain

$$f(\exp_{\bar{p}}(2^{-r}[-\text{grad } f(\bar{p})])) \geq f(\bar{p}) - 2^{-r}\beta\|\text{grad } f(\bar{p})\|^2.$$

The latter inequality is equivalent to

$$-\frac{f(\exp_{\bar{p}}(2^{-r}[-\text{grad } f(\bar{p})])) - f(\bar{p})}{2^{-r}} \leq \beta\|\text{grad } f(\bar{p})\|^2.$$

Thus, letting r go to $+\infty$ we obtain $\|\text{grad } f(\bar{p})\|^2 \leq \beta\|\text{grad } f(\bar{p})\|^2$, which implies $\text{grad } f(\bar{p}) = 0$, i.e., $\bar{p} \in \Omega^*$. Therefore, since $\{p_k\}$ is quasi-Fejér convergent to Ω^* , we conclude from Theorem 2.7 that $\{p_k\}$ converges to \bar{p} and the proof is completed. \square

3.2. Iteration-complexity analysis. In this section, we present an iteration-complexity bound related to the gradient method for minimizing a convex function with Lipschitz continuous gradient with constant $L > 0$. In the following, as an application of Lemma 3.10, we obtain the iteration-complexity bound for the gradient method with Strategy 3.3.

THEOREM 3.12. *Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategy 3.3 for $\beta = 1/2$. Then, for every $N \in \mathbb{N}$, it holds that*

$$(3.12) \quad f(p_N) - f^* \leq \eta L \frac{L_0 d^2(p_0, q) + 2(C_{\rho, \kappa}^q - 1)[f(p_0) - f^*]}{2NL_0}$$

for each $q \in \Omega^*$. As a consequence, given a tolerance $\epsilon > 0$, the number of iterations required to obtain $p_N \in \mathcal{M}$ such that $f(p_N) - f^* < \epsilon$ is bounded by

$$\eta L [L_0 d^2(p_0, q) + 2(C_{\rho, \kappa}^q - 1)[f(p_0) - f^*]] / (2L_0\epsilon) = \mathcal{O}(1/\epsilon).$$

Proof. Take $q \in \Omega^*$. After some algebraic manipulations, Lemma 3.10 implies

$$2t_k(f(p_{k+1}) - f^*) \leq [d^2(p_k, q) - d^2(p_{k+1}, q)] + 2t_k[C_{\rho, \kappa}^q - 1][f(p_k) - f(p_{k+1})]$$

for all $k = 0, 1, \dots$. Using (3.6) and taking into account that $C_{\rho, \kappa}^q \geq 1$, $f(p_{k+1}) - f^* \geq 0$, and $f(p_k) - f(p_{k+1}) \geq 0$ for all $k = 0, 1, \dots$, it follows that

$$\frac{2}{\eta L} [f(p_{k+1}) - f^*] \leq [d^2(p_k, q) - d^2(p_{k+1}, q)] + \frac{2}{L_0} [C_{\rho, \kappa}^q - 1][f(p_k) - f(p_{k+1})].$$

Summing both sides of the above inequality for $k = 0, 1, \dots, N-1$, we obtain

$$\frac{2}{\eta L} \sum_{i=0}^{N-1} [f(p_{i+1}) - f^*] \leq [d^2(p_0, q) - d^2(p_N, q)] + \frac{2}{L_0} [C_{\rho, \kappa}^q - 1][f(p_0) - f(p_N)].$$

Since $\{f(x_k)\}$ is a decreasing sequence, we conclude that

$$\frac{2}{\eta L} N(f(p_N) - f^*) \leq [d^2(p_0, q) - d^2(p_N, q)] + \frac{2}{L_0} [C_{\rho, \kappa}^q - 1][f(p_0) - f(p_N)],$$

which is equivalent to (3.12). The second statement of the theorem follows as an immediate consequence of the first part. \square

We can take a constant step size whenever the Lipschitz constant $L > 0$ is computable, and Theorem 3.12 trivially implies the following result.

THEOREM 3.13. Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategy 3.2 for $t_k = 1/L$ for all $k = 0, 1, \dots$. Then, for every $N \in \mathbb{N}$, it holds that

$$(3.13) \quad f(p_N) - f^* \leq \frac{L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]}{2N}$$

for each $q \in \Omega^*$. As a consequence, given a tolerance $\epsilon > 0$, the number of iterations required by the gradient method to obtain $p_N \in \mathcal{M}$ such that $f(p_N) - f^* < \epsilon$, is bounded by

$$[L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]] / (2\epsilon) = \mathcal{O}(1/\epsilon).$$

We remark that, if $\kappa = 0$, then $\mathcal{C}_{\rho, \kappa}^q = 1$. As a consequence, Theorem 3.13 transforms into [6, Theorem 3.2].

COROLLARY 3.14. Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategy 3.2 for $t_k = 1/L$ for all $k = 0, 1, \dots$. Then, for every $N \in \mathbb{N}$, it holds that

$$(3.14) \quad \min\{\|\text{grad } f(p_k)\| : k = 0, 1, \dots, N\} \leq \frac{2\sqrt{L[L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]]}}{N}$$

for each $q \in \Omega^*$. As a consequence, given a tolerance $\epsilon > 0$, the number of iterations required by the gradient method to obtain $p_N \in \mathcal{M}$ such that $\|\text{grad } f(p_N)\| < \epsilon$ is bounded by $\sqrt{L[L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]]} / \epsilon = \mathcal{O}(1/\epsilon)$.

Proof. Let $N \in \mathbb{N}$. Using the notation $\lceil N/2 \rceil$ for the least integer that is greater than or equal to $N/2$, we have

$$(3.15) \quad f(p_{N+1}) - f^* + \sum_{j=\lceil N/2 \rceil}^N [f(p_j) - f(p_{j+1})] = f(p_{\lceil N/2 \rceil}) - f^*.$$

Thus, combining the last inequality with Theorem 3.13, we conclude that

$$(3.16) \quad f(p_{N+1}) - f^* + \sum_{j=\lceil N/2 \rceil}^N [f(p_j) - f(p_{j+1})] \leq \frac{L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]}{2\lceil N/2 \rceil}.$$

On the other hand, using Lemma 3.7 and considering that $t_k = 1/L$, we obtain

$$\frac{1}{2L} \sum_{j=\lceil N/2 \rceil}^N \|\text{grad } f(p_j)\|^2 \leq \sum_{j=\lceil N/2 \rceil}^N [f(p_j) - f(p_{j+1})] \leq f(p_{\lceil N/2 \rceil}) - f^*.$$

In view of $N/2 \leq \lceil N/2 \rceil$, the above inequality, together with (3.15) and (3.16), yields

$$\frac{1}{2L} \sum_{j=\lceil N/2 \rceil}^N \|\text{grad } f(p_j)\|^2 \leq \frac{L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]}{N}.$$

Therefore,

$$\min\{\|\text{grad } f(p_k)\|^2 : k = \lceil N/2 \rceil, \dots, N\} \leq \frac{4L[L d^2(p_0, q) + 2(\mathcal{C}_{\rho, \kappa}^q - 1)[f(p_0) - f^*]]}{N^2},$$

which implies the desired inequality. The second statement of the corollary follows as an immediate consequence of the first one. \square

We end this section by recalling an iteration-complexity bound for nonconvex functions defined in a general Riemannian manifold, which appeared in [8].

THEOREM 3.15. *Let $\{p_k\}$ be generated by Algorithm 3.1 with Strategy 3.2. Then, for every $N \in \mathbb{N}$, it holds that*

$$\min\{\|\text{grad } f(p_k)\| : k = 0, 1, \dots, N\} \leq \frac{\sqrt{2L(f(p_0) - f^*)}}{\sqrt{N+1}}.$$

As a consequence, given a tolerance $\epsilon > 0$, the number of iterations required to obtain $p_N \in \mathcal{M}$ such that $\|\text{grad } f(p_N)\| < \epsilon$ is bounded by $\mathcal{O}(L(f(p_0) - f^*)/\epsilon^2)$.

Under the assumption of convexity and lower boundedness of the curvature, we conclude that Corollary 3.14 improves on Theorem 3.15. It is worth pointing out that results on the iteration-complexity bound for the gradient method on a Riemannian manifold with nonnegative curvature and in Hadamard manifolds with lower bound curvature appeared in [6, 41, 42]. The results of this section contribute to the systematic study of the iteration complexity of gradient methods in the Riemannian setting.

4. Examples. In this section, we present some examples of functions satisfying the assumptions of our results in the previous sections. In particular, we show that, by endowing the constrained set with a suitable Riemannian metric, a constrained Euclidian optimization problem with nonconvex objective function having a non-Lipschitz gradient can be seen as an unconstrained Riemannian optimization problem with convex objective function having Lipschitz gradient. For use throughout the following subsections we define the positive orthant

$$\mathbb{R}_{++}^n := \{x := (x_1, \dots, x_n)^T \in \mathbb{R}^{n \times 1} : x_i > 0, i = 1, \dots, n\},$$

and denote by \mathbb{P}^n the set of symmetric matrices of order $n \times n$ and \mathbb{P}_{++}^n the cone of symmetric positive definite matrices.

4.1. Examples in the positive orthant. In this section, we present examples in the positive orthant endowed with a new Riemannian metric. To present these examples we need some definitions and results of Riemannian geometry. Endowing \mathbb{R}_{++}^n with the Riemannian metric $\langle \cdot, \cdot \rangle$ defined by $\langle u, v \rangle := u^T G(x)v$ for all $x \in \mathbb{R}_{++}^n$ and $u, v \in T_x \mathbb{R}_{++}^n \equiv \mathbb{R}^n$, where $G: \mathbb{R}_{++}^n \rightarrow \mathbb{P}_{++}^n$ is given by

$$(4.1) \quad G(x) := \text{diag}(x_1^{-2}, \dots, x_n^{-2}) \in \mathbb{R}^{n \times n},$$

we obtain a complete Riemannian manifold with zero curvature, which will be denoted by $\mathcal{M} := (\mathbb{R}_{++}^n, G)$. Let $f: \mathcal{M} \rightarrow \mathbb{R}$ be a twice-differentiable function. We denote by $f'(x)$ and $f''(x)$ the Euclidian gradient and Hessian of f at x , respectively. Thus, (4.1) implies that the Riemannian gradient and Hessian of f are given, respectively, by

$$(4.2) \quad \text{grad } f(x) = \text{diag}(x)^2 f'(x), \quad x \in \mathcal{M},$$

$$(4.3) \quad \text{hess } f(x)v = [\text{diag}(x)^2 f''(x) + \text{diag}(x)\text{diag}(f'(x))]v, \quad x \in \mathcal{M},$$

where $\text{diag}(x) := \text{diag}(x_1, \dots, x_n) \in \mathbb{R}^{n \times n}$. Next we present two examples of convex functions with Lipschitz gradient in $\mathcal{M} := (\mathbb{R}_{++}^n, G)$.

Example 4.1. Consider the function $f: \mathbb{R}_{++}^n \rightarrow \mathbb{R}$ defined by

$$(4.4) \quad f(x) := \sum_{i=1}^n f_i(x_i), \quad f_i(x_i) := -a_i e^{-b_i x_i} + c_i \ln(x_i)^2 + d_i \ln(x_i), \quad i = 1, \dots, n,$$

where $a_i, b_i, d_i \in \mathbb{R}_+$ and $c_i \in \mathbb{R}_{++}$ satisfy $c_i > a_i$. Since f is coercive, it has a minimum. By using (4.4), the first and second derivatives of f at $x \in \mathbb{R}_{++}^n$ are given by $f'(x) := (f'_1(x_1), \dots, f'_n(x_n))$ and $f''(x) := \text{diag}(f''_1(x_1), \dots, f''_n(x_n))$, where

$$(4.5) \quad f'_i(x_i) = a_i b_i e^{-b_i x_i} + 2c_i \frac{\ln(x_i)}{x_i} + \frac{d_i}{x_i}, \quad f''_i(x_i) = -a_i b_i^2 e^{-b_i x_i} + 2c_i \left[\frac{1 - \ln(x_i)}{x_i^2} \right] - \frac{d_i}{x_i^2}$$

for all $i = 1, \dots, n$. Note that $f''_i(1) < 0$ for all $i = 1, \dots, n$, and then f is not Euclidian convex. Using (4.3) and (4.5), the Hessian of f in $\mathcal{M} := (\mathbb{R}_{++}^n, G)$ is given by

$$\text{hess } f(x)v := (a_1 b_1 e^{-b_1 x_1} (x_1 - b_1 x_1^2) + 2c_1, \dots, a_n b_n e^{-b_n x_n} (x_n - b_n x_n^2) + 2c_n)v.$$

Since $c_i > a_i$, we have $a_i b_i e^{-b_i x_i} (x_i - b_i x_i^2) + 2c_i \geq 0$ for all $i = 1, \dots, n$. Hence, by using the definition of the metric, for $v = (v_1, \dots, v_n)^T \in \mathbb{R}^n$ and $x \in \mathbb{R}_{++}$, we have

$$\langle \text{hess } f(x)v, v \rangle = \sum_{i=1}^n [a_i b_i e^{-b_i x_i} (x_i - b_i x_i^2) + 2c_i] \frac{v_i^2}{x_i^2} \geq 0,$$

yielding that f is convex in \mathcal{M} . Since $\|v\| = v^T G(x)v = 1$, we have $v_i^2 \leq x_i^2$, and due to $(a_i b_i e^{-b_i x_i} (x_i - b_i x_i^2) + 2c_i) < a_i + 2c_i$ for all $i = 1, \dots, n$, we obtain

$$\|\text{hess } f(x)v\|^2 = \sum_{i=1}^n [a_i b_i e^{-b_i x_i} (x_i - b_i x_i^2) + 2c_i]^2 \frac{v_i^2}{x_i^2} < \sum_{i=1}^n (a_i + 2c_i)^2, \quad x \in \mathbb{R}_{++}.$$

Therefore, (2.2) and Lemma 2.3 imply $\text{grad } f$ is Lipschitz with $L < \sum_{i=1}^n (a_i + 2c_i)^2$.

Example 4.2. Consider the function $f: \mathbb{R}_{++}^n \rightarrow \mathbb{R}$ defined by

$$(4.6) \quad f(x) := \sum_{i=1}^n f_i(x_i), \quad f_i(x_i) := a_i \ln(x_i^{d_i} + b_i) - c_i \ln(x_i), \quad i = 1, \dots, n,$$

where $a_i, b_i, c_i, d_i \in \mathbb{R}_{++}$ satisfy $c_i < a_i d_i$ and $d_i \geq 2$ for all $i = 1, \dots, n$. The minimizer of f is $x^* = (x_1^*, \dots, x_n^*)$, where $x_i^* = \sqrt[d_i]{b_i c_i / (a_i d_i - c_i)}$, for $i = 1, \dots, n$. Function f in (4.6) is not Euclidian convex. However, by following the same steps as in Example 4.1, we can show that f is convex and has Lipschitz gradient with constant $L < \sum_{i=1}^n a_i^2 d_i^4$ in $\mathcal{M} = (\mathbb{R}_{++}^n, G)$.

We end this subsection by presenting, without giving the details, two more examples of convex functions with Lipschitz gradients in $\mathcal{M} := (\mathbb{R}_{++}^n, G)$.

Remark 4.3. Let $a, b, c \in \mathbb{R}_{++}$. Define $h_1: \mathbb{R}_{++}^n \rightarrow \mathbb{R}$ by $h_1(x) := a \ln(x^T x + b) - c \ln(x_1 \cdots x_n)$, where $nc < 2a$, and $h_2: \mathbb{R}_{++}^n \rightarrow \mathbb{R}$ by $h_2(x) = a \ln((x_1 \cdots x_n)^2 + b) - c \ln(x_1 \cdots x_n)$. By using similar arguments to Example 4.1, we can prove that h_1 and h_2 are also convex with Lipschitz gradient in the Riemannian manifold $\mathcal{M} = (\mathbb{R}_{++}^n, G)$.

4.2. Examples in the SPD matrices cone. In this section, we present examples in the cone of symmetric positive definite matrices with new Riemannian metric. Following Rothaus [32], let $\mathcal{M} := (\mathbb{P}_{++}^n, \langle \cdot, \cdot \rangle)$ be the Riemannian manifold endowed with the Riemannian metric given by

$$(4.7) \quad \langle U, V \rangle := \text{tr}(VX^{-1}UX^{-1}), \quad X \in \mathcal{M}, \quad U, V \in T_X\mathcal{M},$$

where $\text{tr}(X)$ denotes the trace of $X \in \mathbb{P}^n$ and $T_X\mathcal{M} \approx \mathbb{P}^n$. In fact, \mathcal{M} is a Hadamard manifold (see, for example, [21, Theorem 1.2, Page 325]) and its curvature is bound below (see [22]). The *gradient* and *Hessian* of $f : \mathbb{P}_{++}^n \rightarrow \mathbb{R}$ are given by

$$(4.8) \quad \text{grad } f(X) = Xf'(X)X,$$

$$(4.9) \quad \text{hess } f(X)V = Xf''(X) VX + \frac{1}{2}[Vf'(X)X + Xf'(X)V],$$

where $V \in T_X\mathcal{M}$, and $f'(X)$ and $f''(X)$ are the Euclidian gradient and Hessian of f at X , respectively. In the following, we present two examples of convex functions with Lipschitz gradient in $\mathcal{M} := (\mathbb{P}_{++}^n, \langle \cdot, \cdot \rangle)$.

Example 4.4. Consider the function $f : \mathbb{P}_{++}^n \rightarrow \mathbb{R}$ defined by

$$(4.10) \quad f(X) = a \ln(\det(X))^2 - b \ln(\det(X)),$$

where $a, b \in \mathbb{R}_{++}$. The Euclidian gradient and Hessian of f are given, respectively, by

$$(4.11) \quad f'(X) = [2a \ln(\det(X)) - b]X^{-1},$$

$$(4.12) \quad f''(X)V = 2a \text{tr}(X^{-1}V)X^{-1} - [2a \ln(\det(X)) - b]X^{-1}VX^{-1}$$

for all $X \in \mathbb{P}_{++}^n$ and $V \in \mathbb{P}^n$. It follows from (4.11) that each $X \in \mathcal{M}$ satisfying $\det X = e^{b/(2a)}$ is a critical point of f . Thus, letting $V = I_n$ and $X = tI_n$ with $t \in \mathbb{R}_{++}$ in (4.12) we obtain $f''(tI_n)I_n = [2ant^{-2} - 2an \ln t + b]I_n$. Thus, $f''(tI_n)$ is not positive definite for t sufficiently large. Hence, f is not Euclidian convex. Moreover, f'' is not bounded and, consequently, f' is not Lipschitz. On the other hand, combining (4.9), (4.11), and (4.12), we obtain, after some calculations, that

$$(4.13) \quad \text{hess } f(X)V = 2a \text{tr}(X^{-1}V)X, \quad \langle \text{hess } f(X)V, V \rangle = 2a \text{tr}(X^{-1}V)^2 \geq 0$$

for all $X \in \mathcal{M}$ and $V \in T_X\mathcal{M}$. Thus, f is convex in \mathcal{M} . Moreover, (4.7), together with (4.13), yields $\|\text{hess } f(X)V\| = 2a \text{tr}(X^{-1}V)$ for all $X \in \mathcal{M}$ and $V \in T_X\mathcal{M}$. If we assume that $\|V\|^2 = \text{tr}(VX^{-1}VX^{-1}) = 1$, then $\text{tr}(X^{-1}V) \leq \sqrt{n}$. Hence,

$$\|\text{hess } f(X)V\| \leq 2a\sqrt{n}, \quad X \in \mathcal{M}, \quad V \in T_X\mathcal{M}, \quad \|V\| = 1.$$

Therefore, (2.2) and Lemma 2.3 imply that $\text{grad } f$ is Lipschitz with constant $L \leq 2a\sqrt{n}$.

Example 4.5. Consider the function $f : \mathbb{P}_{++}^n \rightarrow \mathbb{R}$ defined by

$$(4.14) \quad f(X) = a \ln(\det(X)^{b_1} + b_2) - c \ln(\det X),$$

where $a, b_1, b_2, c \in \mathbb{R}_{++}$ with $c < ab_1$. Function f in (4.14) is not Euclidian convex. On the other hand, by using similar arguments as in Example 4.4, we can see that f is convex and has Lipschitz gradient with constant $L < ab_1^2 n$ in $\mathcal{M} = (\mathbb{P}_{++}^n, \langle \cdot, \cdot \rangle)$.

5. Numerical experiments. In this section, we present some numerical experiments to illustrate the behavior of the Riemannian gradient method for minimizing convex functions in the positive orthant and the cone of symmetric positive definite matrices. We implemented Algorithm 3.1 with Strategies 3.2–3.4, and tested it on the examples in section 4. Additionally, we apply the method to compute the Riemannian center of mass, which is a specific instance of a geometric mean for points in a Riemannian manifold. In due course, we will describe this problem in more detail.

For the positive orthant, the *exponential mapping* $\exp_x: T_x\mathcal{M} \rightarrow \mathcal{M}$ in the Riemannian manifold $\mathcal{M} := (\mathbb{R}_{++}^n, G)$ is assigned by

$$(5.1) \quad \exp_x(v) = \left(x_1 e^{\frac{v_1}{x_1}}, \dots, x_n e^{\frac{v_n}{x_n}} \right)$$

for each $v := (v_1, \dots, v_n)^T \in \mathbb{R}^{n \times 1}$ and $x := (x_1, \dots, x_n)^T \in \mathbb{R}_{++}^n$; see [29]. By using the gradient in (4.2) and the definition of the metric, we obtain

$$\|\text{grad } f(x)\|^2 = \text{grad } f(x)^T G(x) \text{grad } f(x) = \sum_{i=1}^n \left[x_i \frac{\partial f}{\partial x_i}(x) \right]^2$$

for each $x := (x_1, \dots, x_n) \in \mathcal{M}$. Considering the cone of symmetric positive definite matrices, the *exponential mapping* $\exp_X: T_X\mathcal{M} \rightarrow \mathcal{M}$ in the Riemannian manifold $\mathcal{M} := (\mathbb{P}_{++}^n, \langle \cdot, \cdot \rangle)$ is given by

$$(5.2) \quad \exp_X(V) = X^{1/2} e^{(X^{-1/2} V X^{-1/2})} X^{1/2}$$

for each $V \in \mathbb{P}^n$ and $X \in \mathbb{P}_{++}^n$. By using (4.8), for each $X \in \mathcal{M}$, we have $\|\text{grad } f(X)\|^2 = \text{tr}([X f'(X)]^2)$. In both cases, although (1.1) is a constrained optimization problem, by (5.1) and (5.2), Algorithm 3.1 generates only feasible points without using projections or any other procedure to retain the feasibility. Hence, problem (1.1) can be seen as unconstrained Riemannian optimization problem.

Our numerical experience indicates that it is advantageous to perform a reasonably stringent line search. Therefore, we used $\beta = 1/2$ for Strategies 3.3 and 3.4. Additionally, we set $L_0 = 1$ and $\eta = 2$ for Strategy 3.3. We stopped the execution of Algorithm 3.1 at p_k , declaring convergence if

$$\|f'(p_k)\|_\infty \leq 10^{-5}.$$

Since, by (4.2) and (4.8), $\text{grad } f(p_k) = 0$ if only if $f'(p_k) = 0$, this is a reasonable stopping criterion. The maximum number of iterations allowed was set to 1000. Codes were written in MATLAB and are freely available at <https://orizon.mat.ufg.br/>.

5.1. Academic problems. We begin the numerical experiments by testing the Riemannian gradient method on the problems of minimizing the functions in the examples in section 4. We call these Problems 1, 2, 3, and 4, respectively.

5.1.1. Academic problems in the positive orthant. In this section, we compare the performance of the Riemannian and Euclidian gradient methods on Problems 1 and 2. We consider Algorithm 3.1 with Strategy 3.4 and implement the Euclidian gradient method using the Armijo rule with the same algorithmic parameters. It is worth mentioning that, in principle, the Euclidian method can generate iterates outside the positive orthant. Thus, in order to keep the feasibility, in each iteration we simply determine the maximum step size to remain within the feasible set and

perform a convenient line search by shrinking the step size until the Armijo condition is satisfied.

We generated several instances of problems 1 and 2 by considering functions (4.4) and (4.6), respectively, with $n = 100$ and different parameters. In all cases, for each $i = 1, \dots, n$, we set parameters a_i with the same value, and equivalently for parameters b_i , c_i , and d_i .

Problem 1. First, parameters a_i , b_i , and d_i were randomly generated between 0 and 10. Then, in order to guarantee that $c_i > a_i$, we randomly generated parameters c_i between $1.1a_i$ and $5.0a_i$. All problems were solved 100 times using starting points from a uniform random distribution inside the box $[0, 20]^n$. For each method, Table 1 shows the percentage of runs that reached a critical point (%), the average number of iterations (it), and function evaluations (nfev) of the successful runs.

TABLE 1

Parameters of function (4.4) as well as the performance of the Riemannian and Euclidian gradient methods.

#	a_i	b_i	c_i	d_i	Riemannian gradient method			Euclidian gradient method		
					%	it	nfev	%	it	nfev
1	3.77	8.17	11.10	5.92	100.0	14.1	85.5	100.0	72.3	255.4
2	7.88	5.49	17.95	3.01	100.0	21.1	148.5	100.0	56.9	208.2
3	8.96	1.72	42.11	7.18	100.0	17.0	137.1	100.0	56.0	203.3
4	3.14	1.30	13.77	9.32	100.0	9.0	55.0	100.0	76.3	232.6
5	5.49	1.72	6.82	0.83	100.0	10.0	51.0	100.0	65.1	227.3
6	4.59	4.25	13.31	8.11	100.0	11.0	67.0	100.0	71.2	228.5
7	2.10	3.80	4.31	0.10	100.0	21.2	107.0	100.0	54.1	184.7
8	8.69	7.47	28.54	4.77	100.0	8.0	57.1	100.0	61.1	255.3
9	9.85	2.24	44.60	0.57	100.0	16.0	129.0	100.0	52.0	201.9
10	2.60	1.71	9.65	2.07	100.0	18.0	109.2	100.0	57.1	185.6
11	6.03	1.40	13.57	8.94	100.0	9.0	55.0	100.0	79.2	238.1
12	5.71	4.99	9.37	3.22	100.0	20.1	121.7	100.0	59.2	191.2
13	1.38	6.07	6.78	4.86	100.0	9.0	46.1	100.0	73.5	219.6
14	2.22	0.24	5.58	9.04	100.0	14.0	71.0	100.0	141.8	408.6
15	4.19	6.24	7.73	9.48	100.0	7.0	36.0	100.0	105.8	315.4
16	8.27	2.42	10.96	3.02	100.0	17.0	103.0	100.0	66.3	237.4
17	4.72	0.64	19.35	0.62	100.0	18.0	127.0	100.0	55.6	204.1
18	2.99	1.63	11.15	6.44	100.0	14.0	85.1	100.0	75.8	250.8

As can be seen, the Riemannian gradient method is clearly more efficient than the Euclidian gradient method in this set of problems. In *all* 18 problem instances considered, the Riemannian version required fewer iterations and function evaluations. Overall, on average, the Riemannian gradient method performed 19.8% of iterations and 37.5% of the function evaluations required by the Euclidian method.

Figure 1(a) shows a typical behavior of the methods for problem 1. This case corresponds to $n = 2$, $a_i = 1$, $b_i = c_i = d_i = 2$ for $i = 1, 2$, and the initial point $p_0 = [5, 1]^T$. The stopping criterion was satisfied with 4 and 14 iterations for the Riemannian and Euclidian gradient methods, respectively. The *zigzag* path of the Euclidian gradient method can be seen clearly. In contrast, the Riemannian method rapidly approaches the minimizer. In Figure 1(b), the sup-norm of the Euclidian gradient is displayed as a function of the iteration number, which clearly shows the distinction between the methods. While the Euclidian method required 10 iterations for $\|f'(p_k)\|_\infty$ to reach the order of 10^{-2} , the Riemannian algorithm required only 2 iterations.

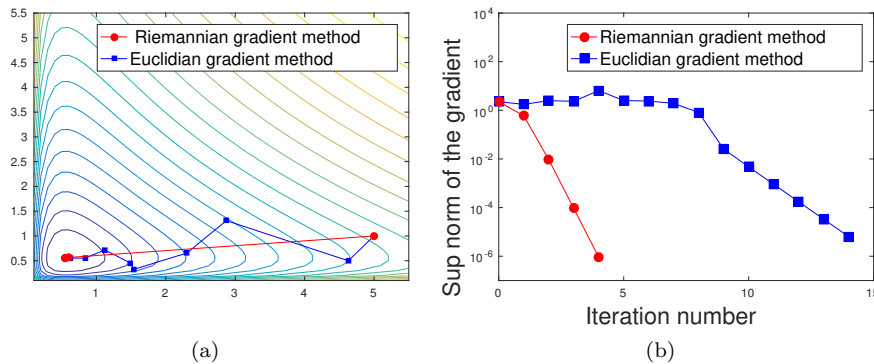


FIG. 1. (a) A typical behavior of the Riemannian and Euclidian gradient methods; a zigzag pattern appears for the Euclidian algorithm. (b) Sup-norm of the Euclidian gradients per iteration.

Problem 2. We tested the algorithms on a set of 100 instances of problem 2. We randomly generated parameters a_i and b_i between 0 and 10, parameters d_i between 2 and 10, and a constant μ_i belonging to the interval $(0,1)$. Then, we set $c_i = \mu_i a_i d_i$, fulfilling the conditions $c_i < a_i d_i$ and $d_i \geq 2$ for all $i = 1, \dots, n$. As for problem 1, each instance was solved 100 times using starting points from a uniform random distribution inside the box $[0, 20]^n$. The results are given in the following form: for each problem instance, Figure 2(a) displays the average number of iterations, and Figure 2(b) shows the average number of function evaluations. As a matter of aesthetics, the graphs were organized in an increasing way.

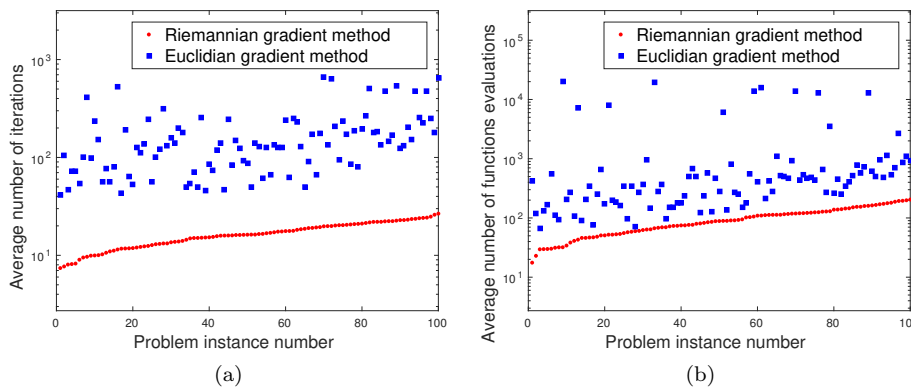


FIG. 2. (a) Average number of iterations and (b) average number of function evaluations required for each of 100 instances of problem 2 for the Riemannian and Euclidian gradient methods.

Figure 2 shows that the Riemannian gradient method required fewer iterations and function evaluations than the Euclidian gradient method in *all* problem instances. In terms of percentages, on average, the Riemannian algorithm performed 9.7% and 5.6% of the number of iterations and function evaluations required by the Euclidian algorithm, respectively.

The results of this section allow us to conclude that there are problems for which the introduction of a suitable metric makes it possible to explore their geometric and algebraic structures, resulting in a large reduction in the computational cost of

obtaining their solution. In fact, by introducing a suitable Riemannian metric, a constrained optimization problem with nonconvex objective function and non-Lipschitz gradient can be transformed into an optimization problem with convex objective function and Lipschitz gradient.

5.1.2. Academic problems in the SPD matrices cone. In this section we illustrate the practical applicability of the Riemannian gradient method for minimizing convex functions in the cone of symmetric positive definite matrices. We used problem 3 to test the Riemannian gradient method by varying the dimension and the domain of the starting points, and problem 4 to compare the different line search strategies. For problem 3, we adopted Strategy 3.4.

Problem 3. We set $a = b = 1$ in function (4.10). In the first set of tests, we assigned the following values to the dimension: $n = 10, 20, 50, 100$, and 150 . For each specific value of n , we ran the Riemannian gradient method 100 times using random starting points with eigenvalues belonging to the interval $(0, 20)$. In the second set of tests, we set $n = 50$ and varied the interval that contains the eigenvalues of the starting points. Again, for each combination, the method was run 100 times using random starting points. The results for the first and second set of tests are shown in parts (a) and (b) of Table 2, respectively. First column of Table 2(a) informs the considered dimension, while the first column of Table 2(b) contains the interval for the eigenvalues of the starting points. Column headings “%,” “it,” and “nfev” are as defined for Table 1.

TABLE 2

Performance of the Riemannian gradient method in problem 3 varying (a) the dimension, (b) the domain of the starting points.

n	%	it	nfev
10	100.0	18.2	110.2
20	100.0	19.9	140.4
50	100.0	14.2	114.9
100	100.0	15.2	138.2
150	100.0	27.1	271.5

(a)

$\lambda_i(X_0)$	%	it	nfev
(0, 10)	98.0	14.2	114.6
(0, 100)	99.0	14.6	117.6
(0, 500)	99.0	15.0	121.0
(0, 1000)	100.0	15.1	121.6
(0, 2000)	100.0	15.2	122.4

(b)

Table 2 highlights that the Riemannian gradient method was robust with respect to the dimension and to the choice of starting point. Furthermore, except for the case when $n = 150$, it was not very sensitive to the variation in dimension or to the domain of starting points.

For comparison, we implemented and tested the Euclidian method in problem 3. For $n = 5$ (respectively, $n = 10$), 15 (respectively, 96) out of the 100 starting points considered resulted in an iteration history that reached the maximum number of iterations allowed. Finally, we observe that, by using (5.2) and the function (4.10), the Riemannian and Euclidian gradient iterations become

$$X_{k+1} = [\det(X_k)^{2a} e^b]^{-t_k} X_k, \quad k = 0, 1, \dots,$$

and

$$X_{k+1} = X_k - t_k [2a \ln(\det(X_k)) - b] X_k^{-1}, \quad k = 0, 1, \dots,$$

respectively, where the step size $t_k > 0$ is computed according to the adopted line search strategy. Thus, we can see that the Riemannian gradient iterations are simpler and can be performed with a lower computational cost.

Problem 4. We set $n = 100$, $a = b_1 = b_2 = 1$, and $c = 0.5$ in function (4.14), fulfilling $c < ab_1$. We tested the Riemannian gradient method with each of the three strategies by running each combination 100 times using random starting points with eigenvalues belonging to the interval $(0, 20)$. The results in Table 3 are given for the same variables as in the previous tables.

TABLE 3
Performance of the Riemannian gradient method with the different line search strategies.

Strategy 3.2			Strategy 3.3			Strategy 3.4		
%	it	nfev	%	it	nfev	%	it	nfev
100.0	452.5	453.5	99.0	15.3	21.3	100.0	15.3	70.9

For Strategy 3.2, since the Lipschitz gradient constant satisfies $L < ab_1^2 n$, we used the Lipschitz step size $t_k = 1/(ab_1^2 n) < 1/L$ for all $k = 1, 2, \dots$. Overall, as can be seen in Table 3, the Riemannian method with Lipschitz step sizes is clearly the least efficient, requiring an exceedingly large number of iterations. In this case the method performs one function evaluation per iteration. The poor performance is due to the short step sizes in all iterations. On the other hand, we point out that the efficiency of the Riemannian gradient method with Lipschitz step size is closely related to an accurate estimate of the Lipschitz gradient constant.

Remark 3.6 helps to explain the results of Table 3 for Strategies 3.3 and 3.4. Regardless of the starting point, Algorithm 3.1 performed exactly the same number of iterations with both strategies. Additionally, in a typical run, the step sizes were nonincreasing. Therefore, overall, by Remark 3.6, the adaptive scheme in Strategy 3.3 required fewer function evaluations per iteration than the Armijo line search of Strategy 3.4.

Despite the simple line search mechanisms employed here, the numerical results indicate that, as expected, the efficient implementation of line search algorithms can significantly improve the Riemannian gradient method.

5.2. The Riemannian center of mass. The Riemannian center of mass and so-called Karcher mean are a specific instance of a geometric mean for points in Riemannian manifolds. They have several practical applications and have appeared in many papers; we refer the reader to [7, 20, 19, 35] and the references therein. It is worth mentioning that studies relating to the center of mass using the Riemann–Finsler geometry are also presented in [20], and an interesting practical model is provided; however, unlike what we do here, genetic algorithms are applied to locate the center of mass in certain situations.

5.2.1. The center of mass on the SPD matrices cone. Denote by $\|\cdot\|_F$ the Frobenius norm associated to the inner product $\langle U, V \rangle_F := \text{tr}(VU)$ for all $U, V \in \mathbb{P}_{++}^n$. Let d be the Riemannian distance defined in $\mathcal{M} := (\mathbb{P}_{++}^n, \langle \cdot, \cdot \rangle)$, i.e.,

$$(5.3) \quad d(A, X) = \|\ln(X^{-1/2}AX^{-1/2})\|_F, \quad A, X \in \mathbb{P}_{++}^n;$$

see [29]. The Karcher mean of m positive definite matrices $A_1, \dots, A_m \in \mathbb{P}_{++}^n$ is the unique solution of the optimization problem

$$(5.4) \quad \min \left\{ f(x) := \frac{1}{2} \sum_{j=1}^m \|\ln(X^{-1/2}A_jX^{-1/2})\|_F^2 : X \in \mathbb{P}_{++}^n \right\}.$$

Indeed, f is a strong convex function in \mathcal{M} due to the square of the distance (5.3) being strongly convex in \mathcal{M} ; see, for example, [12]. Since f is a strong convex function,

all sublevel sets of f are bounded. As a consequence, f has Riemannian Lipschitz gradient on each sublevel set of f . Finally, we remark that (5.3) is not a Euclidian convex function. By [19] and using (4.8), we conclude that

$$(5.5) \quad \text{grad } f(X) = \sum_{i=1}^m X^{1/2} \ln(X^{1/2} A_i^{-1} X^{1/2}) X^{1/2}.$$

Thus, by using (5.2) and (5.5), the Riemannian gradient iteration for solving (5.4) is

$$X_{k+1} = X_k^{1/2} e^{-t_k \sum_{i=1}^n \ln(X_k^{1/2} A_i^{-1} X_k^{1/2})} X_k^{1/2}, \quad k = 0, 1, \dots;$$

see, for example, [42].

In [2], Afsari, Tron, and Vidal studied the convergence of the Riemannian gradient method with a Lipschitz step size for the center of mass problem in a manifold with curvature bounded from above and below. The step size is defined from a local estimate for the Lipschitz gradient constant. Consider problem (5.4), and let $r > 0$ be such that $A_1, \dots, A_m \subset B(X_0, r)$, where $B(X_0, r)$ is the open ball with center X_0 and radius r . Afsari, Tron, and Vidal showed that it is possible to achieve convergence with $t_k = t$ for all $k = 0, 1, \dots$, where $t \in (0, 2\bar{t})$ and

$$(5.6) \quad \bar{t} = \frac{1}{4r \coth(4r)}.$$

Recently, Bento et al. [5] extended the convergence of the gradient method to the Hadamard setting for continuously differentiable functions that satisfy the Kurdyka–Łojasiewicz inequality. In particular, they proposed a Riemannian gradient method with Armijo line search for problem (5.4). Basically, their proposal coincides with Algorithm 3.1 with Strategy 3.4.

We tested Algorithm 3.1 with each strategy on a set of 200 randomly generated problems (5.4) with $n = 200$ and $m = 5, 10, 20$, or 50. For each value of m we considered 50 problem instances. Let us clarify how matrix A was defined. First, we randomly generated an orthonormal matrix U and a diagonal matrix D with elements belonging to $(0, 100)$. Then, we set $A = UDU^T$, ensuring that $A \in \mathbb{P}_{++}^n$. Given a problem instance with data $A_1, \dots, A_m \in \mathbb{P}_{++}^n$, we defined the starting point X_0 as the geometric mean given by

$$X_0 := \exp\left(\frac{1}{m} \sum_{i=1}^m \ln(A_i)\right);$$

see, for example, [3]. For Strategy 3.2 the Lipschitz step size t was defined according to [2]. We set $t = 1.99\bar{t}$, where \bar{t} is given by (5.6). Radius r can be calculated by computing the maximum distance of X_0 to each matrix A_i , $i = 1, \dots, m$. Numerical comparisons are given in Figure 3 using performance profiles [14]. We adopted the number of function evaluations and CPU time as performance measurements.

As can be seen, Algorithm 3.1 with Strategy 3.3 is the most efficient on the chosen set of test problems. The respective efficiencies of the methods are 25.0% (respectively, 24.0%), 75.0% (respectively, 76.0%), and 0.0% (respectively, 0.0%), considering the number of function evaluations (respectively, CPU time) as performance measurement. The efficiency of Algorithm 3.1 with Strategy 3.4 is 0.0% because Strategy 3.3 outperformed Strategy 3.4 in all instances considered. Curiously, m is equal to 20

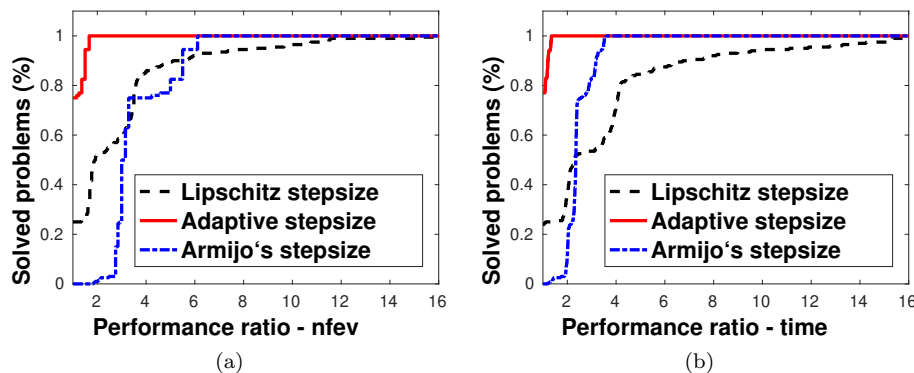


FIG. 3. Performance profile comparing the Riemannian gradient method with different line search strategies using as performance measurement (a) number of function evaluations, (b) CPU time.

TABLE 4

Efficiency and robustness of Algorithm 3.1 with different line search strategies on a set of 200 randomly generated Riemannian center of mass problems.

	Efficiency (nfev – CPU time) (%)	Robustness (%)
Strategy 3.2	25.0 – 24.0	99.5
Strategy 3.3	75.0 – 76.0	100.0
Strategy 3.4	0.0 – 0.0	100.0

in all problems for which Strategy 3.2 was the most efficient. Robustness values are 99.5%, 100.0%, and 100.0%, respectively; see Table 4. Algorithm 3.1 with Strategy 3.2 reached the maximum permitted number of iterations in only one problem instance.

The similarity of parts (a) and (b) in Figure 3 suggests that the number of function evaluations is a good indicator of performance. Indeed, evaluating function f is computationally expensive, since it involves inverting X and computing m matrix logarithms. This implies that line search schemes must be carefully formulated for the center of mass problem. Overall, the naive implementation of the Armijo line search in Strategy 3.4 was overcome by the method with Lipschitz step size. On the other hand, the results indicate that the adaptive search proposed in Strategy 3.3 is a promising scheme worth consideration.

5.2.2. The center of mass on the positive orthant. Let $\mathcal{M} := (\mathbb{R}_{++}^n, G)$ be the Riemannian manifolds defined in section 4.1 and d be the associated Riemannian distance. Hence, we have

$$(5.7) \quad d^2(y, x) = \sum_{i=1}^n \ln^2 \left(\frac{y_i}{x_i} \right), \quad y = (y_1, \dots, y_n), \quad x = (x_1, \dots, x_n) \in \mathbb{R}_{++}^n.$$

The center of mass of m points $w^1, \dots, w^m \in \mathbb{R}_{++}^n$ is the unique solution of the optimization problem

$$(5.8) \quad \min \left\{ f(x) := \frac{1}{2} \sum_{j=1}^m d^2(w^j, x) : x \in \mathbb{R}_{++}^n \right\}.$$

Since the square of the distance (5.7) is strongly convex in \mathcal{M} , f is a strong convex function in \mathcal{M} ; see, for example, [12]. By using (4.2), we conclude that

$$\text{grad } f(x) = \left(x_1 \sum_{j=1}^m \ln \left(\frac{x_1}{w_1^j} \right), \dots, x_n \sum_{j=1}^m \ln \left(\frac{x_n}{w_n^j} \right) \right),$$

where $x = (x_1, \dots, x_n) \in \mathbb{R}_{++}^n$. Problem (5.8) has a closed solution

$$x^* = (x_1^*, \dots, x_n^*) \in \mathbb{R}_{++}^n$$

given by

$$x_i^* = \left(\prod_{j=1}^m w_i^j \right)^{\frac{1}{m}}$$

for all $i = 1, \dots, m$. Indeed, direct calculations show that $\text{grad } f(x^*) = 0$.

Due to the closed-form solution, we use problem (5.8) to illustrate the results on the iteration-complexity bound of section 3.2. We consider the Riemannian gradient algorithm with Lipschitz step size. Note that the set of positive definite diagonal matrices can be identified with \mathbb{R}_{++}^n . Thus, problem (5.8) can be seen as a particular case of problem (5.4) for positive definite diagonal matrices. Given $w^1, \dots, w^m \in \mathbb{R}_{++}^n$ and defining $A_i = \text{diag}(w^i)$ for all $i = 1, \dots, m$, we defined the Lipschitz step size as in section 5.2.

We set $n = 100$, $m = 5$ and randomly generated the elements of w^1, \dots, w^m and initial point x_0 from a uniform distribution on $(0, 100)$. The computed Lipschitz step size was set to $t \approx 0.06$. The Riemannian gradient algorithm stopped declaring “solution was found” after 30 iterations. Parts (a) and (b) of Figure 4 report the function values of the left- and right-hand sides of inequalities (3.13) and (3.14), respectively.

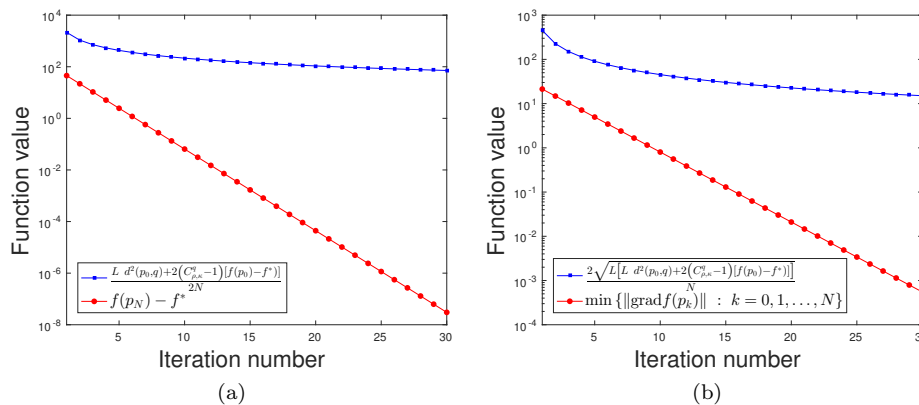


FIG. 4. Iteration-complexity bound for the Riemannian gradient method with Lipschitz step size related to (a) objective function value (Theorem 3.13), (b) norm of the Riemannian gradient (Corollary 3.14).

As can be seen in Figure 4, the iteration-complexity bounds related to the objective function value and the norm of the Riemannian gradient are always respected; see Theorem 3.13 and Corollary 3.14. This illustrates the practical reliability of our iteration-complexity results.

6. Conclusions. In this paper, the behavior of the gradient method for convex optimization problems on Riemannian manifolds with lower bounded sectional curvature was analyzed. We considered three different finite procedures for determining the step size, namely, constant step size, adaptive procedure, and Armijo's procedure. As far as we know, the full convergence of the sequence generated by this method with these three strategies is a new contribution to the literature, and adds important results on the available convergence theories. In addition, under mild assumptions, we showed that the iteration-complexity bound related to the method is $\mathcal{O}(1/\epsilon)$ for finding a point $p_N \in \mathcal{M}$ such that $f(p_N) - f^* < \epsilon$. The numerical experiments provided illustrate the effectiveness of the method in this new setting and certify the conclusions suggested by the theoretical results. Despite the simple line search mechanisms employed here, the numerical results indicate that, as expected, efficient implementation of line search algorithms can significantly improve the Riemannian gradient method. In particular, the effectiveness of the method to find the Riemannian center of mass and the so-called Karcher mean is shown, indicating that the adaptive procedure is a promising scheme that is worth considering. For this reason, it would be interesting to analyze stochastic versions of the gradient method by using adaptive procedures.

We remark that the assumption of boundedness from below of the sectional curvature is just sufficient to obtain the convergence of the gradient method. Indeed, in [2] it was shown that all continuous convex functions on a complete manifold with finite volume are constants. Therefore, letting \mathcal{M} be a Riemannian manifold with sectional curvature unbounded from below and finite volume, we conclude that, for any differentiable convex function $f: \mathcal{M} \rightarrow \mathbb{R}$, the gradient method applied to a minimizer converges trivially to its minimizer (since all convex function f is constant). Note that the set of Riemannian manifolds with sectional curvature unbounded from below and finite volume is nonempty. In fact, a two-dimensional manifold with no lower bound for Gaussian curvature and finite area is presented in [1, Page 457]. One more natural geometrical assumption is the boundedness from below of the Ricci curvature (see [43] and also [10]). One interesting question would then be: is it possible to obtain the results of the present paper by assuming only that Ricci curvature is bounded from below? An affirmative answer to this question would increase the applicability domain of the method. Finally, we expect this paper to contribute to the development of studies of optimization methods in the Riemannian setting, including answers to the above questions.

Acknowledgments. The authors would like to thank the anonymous referees for their reading and positive comments, which helped to improve the presentation of the paper. In particular, we thank one of the referees for drawing our attention to the interesting question about Ricci curvature.

REFERENCES

- [1] P.-A. ABSIL, R. MAHONY, AND R. SEPULCHRE, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, 2008.
- [2] B. AFSARI, R. TRON, AND R. VIDAL, *On the convergence of gradient descent for finding the Riemannian center of mass*, SIAM J. Control Optim., 51 (2013), pp. 2230–2260.
- [3] T. ANDO, C.-K. LI, AND R. MATHIAS, *Geometric means*, Linear Algebra Appl., 385 (2004), pp. 305–334.
- [4] A. BECK AND M. TEOULLE, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.
- [5] G. C. BENTO, S. D. B. BITAR, J. X. CRUZ NETO, P. R. OLIVEIRA, AND J. C. SOUZA, *The Steepest Descent Method for Computing Riemannian Center of Mass on Hadamard Manifolds*,

- preprint, 2017; available at <https://www.researchgate.net/publication/317004835>.
- [6] G. C. BENTO, O. P. FERREIRA, AND J. G. MELO, *Iteration-complexity of gradient, subgradient and proximal point methods on Riemannian manifolds*, J. Optim. Theory Appl., 173 (2017), pp. 548–562.
 - [7] D. A. BINI AND B. IANNAZZO, *Computing the Karcher mean of symmetric positive definite matrices*, Linear Algebra Appl., 438 (2013), pp. 1700–1710.
 - [8] N. BOUMAL, P.-A. ABSIL, AND C. CARTIS, *Global Rates of Convergence for Nonconvex Optimization on Manifolds*, preprint, <https://arxiv.org/abs/1605.08101>, 2016.
 - [9] R. BURACHIK, L. M. G. DRUMMOND, A. N. IUSEM, AND B. F. SVAITER, *Full convergence of the steepest descent method with inexact line searches*, Optimization, 32 (1995), pp. 137–146.
 - [10] T. H. COLDING, *Aspects of Ricci curvature*, in Comparison Geometry, Math. Sci. Res. Inst. Publ. 30, Cambridge University Press, Cambridge, 1997, pp. 83–98.
 - [11] J. X. DA CRUZ NETO, L. L. DE LIMA, AND P. R. OLIVEIRA, *Geodesic algorithms in Riemannian geometry*, Balkan J. Geom. Appl., 3 (1998), pp. 89–100.
 - [12] J. X. DA CRUZ NETO, O. P. FERREIRA, AND L. R. LUCAMBIO PÉREZ, *Contributions to the study of monotone vector fields*, Acta Math. Hungar., 94 (2002), pp. 307–320.
 - [13] M. P. DO CARMO, *Riemannian Geometry*, Mathematics: Theory & Applications, Birkhäuser, Boston, MA, 1992.
 - [14] E. D. DOLAN AND J. J. MORÉ, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), pp. 201–213.
 - [15] P. EBERLEIN, *Lattices in spaces of nonpositive curvature*, Ann. of Math. (2), 111 (1980), pp. 435–476.
 - [16] A. EDELMAN, T. A. ARIAS, AND S. T. SMITH, *The geometry of algorithms with orthogonality constraints*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 303–353.
 - [17] O. P. FERREIRA, A. N. IUSEM, AND S. Z. NÉMETH, *Concepts and techniques of optimization on the sphere*, TOP, 22 (2014), pp. 1148–1170.
 - [18] D. GABAY, *Minimizing a differentiable function over a differential manifold*, J. Optim. Theory Appl., 37 (1982), pp. 177–219.
 - [19] B. JEURIS, R. VANDEBRIL, AND B. VANDEREYCKEN, *A survey and comparison of contemporary algorithms for computing the matrix geometric mean*, Electron. Trans. Numer. Anal., 39 (2012), pp. 379–402.
 - [20] A. KRISTÁLY, G. MOROȘANU, AND A. RÓTH, *Optimal placement of a deposit between markets: Riemann–Finsler geometrical approach*, J. Optim. Theory Appl., 139 (2008), pp. 263–276.
 - [21] S. LANG, *Fundamentals of Differential Geometry*, Grad. Texts in Math. 191, Springer, New York, 1999.
 - [22] C. LENGLET, M. ROUSSON, R. DERICHE, AND O. FAUGERAS, *Statistics on the manifold of multivariate normal distributions: Theory and application to diffusion tensor MRI processing*, J. Math. Imaging Vision, 25 (2006), pp. 423–444.
 - [23] C. LI, B. S. MORDUKHOVICH, J. WANG, AND J.-C. YAO, *Weak sharp minima on Riemannian manifolds*, SIAM J. Optim., 21 (2011), pp. 1523–1560.
 - [24] C. LI AND J.-C. YAO, *Variational inequalities for set-valued vector fields on Riemannian manifolds: Convexity of the solution set and the proximal point algorithm*, SIAM J. Control Optim., 50 (2012), pp. 2486–2514.
 - [25] D. G. LUENBERGER, *The gradient projection method along geodesics*, Management Sci., 18 (1972), pp. 620–631.
 - [26] J. H. MANTON, *A framework for generalising the Newton method and other iterative methods from Euclidean space to manifolds*, Numer. Math., 129 (2015), pp. 91–125.
 - [27] Y. NESTEROV, *Introductory Lectures on Convex Optimization: A Basic Course*, Appl. Optim. 87, Kluwer Academic, Boston, MA, 2004.
 - [28] Y. NESTEROV, *Gradient methods for minimizing composite functions*, Math. Program., 140 (2013), pp. 125–161.
 - [29] Y. E. NESTEROV AND M. J. TODD, *On the Riemannian geometry defined by self-concordant barriers and interior-point methods*, Found. Comput. Math., 2 (2002), pp. 333–361.
 - [30] T. RAPCSÁK, *Smooth nonlinear optimization in R^n* , Nonconvex Optim. Appl. 19, Kluwer Academic, Dordrecht, 1997.
 - [31] M. RAYDAN, *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7 (1997), pp. 26–33.
 - [32] O. S. ROTHBAUS, *Domains of positivity*, Abh. Math. Semin. Univ. Hambg., 24 (1960), pp. 189–235.
 - [33] T. SAKAI, *Riemannian Geometry*, Transl. Math. Monogr. 149, American Mathematical Society, Providence, RI, 1996.
 - [34] S. T. SMITH, *Optimization techniques on Riemannian manifolds*, in Hamiltonian and Gradient

- Flows, Algorithms and Control, Fields Inst. Commun. 3, Amer. Math. Soc., Providence, RI, 1994, pp. 113–136.
- [35] S. SRA AND R. HOSSEINI, *Conic geometric optimization on the manifold of positive definite matrices*, SIAM J. Optim., 25 (2015), pp. 713–739.
 - [36] C. UDRIȘTE, *Convex Functions and Optimization Methods on Riemannian Manifolds*, Math. Appl. 297, Kluwer Academic, Dordrecht, 1994.
 - [37] X. WANG, C. LI, J. WANG, AND J.-C. YAO, *Linear convergence of subgradient algorithm for convex feasibility on Riemannian manifolds*, SIAM J. Optim., 25 (2015), pp. 2334–2358.
 - [38] X. M. WANG, C. LI, AND J. C. YAO, *Subgradient projection algorithms for convex feasibility on Riemannian manifolds with lower bounded curvatures*, J. Optim. Theory Appl., 164 (2015), pp. 202–217.
 - [39] S. T. YAU, *Non-existence of continuous convex functions on certain Riemannian manifolds*, Math. Ann., 207 (1974), pp. 269–270.
 - [40] Y.-X. YUAN, *Step-sizes for the Gradient Method*, in Third International Congress of Chinese Mathematicians (Part 2), AMS/IP Stud. Adv. Math. 42.2, American Mathematical Society, Providence, RI, 2008, pp. 785–796.
 - [41] H. ZHANG, S. J. REDDI, AND S. SRA, *Fast Stochastic Optimization on Riemannian Manifolds*, preprint, <http://arxiv.org/abs/1605.07147>, 2016.
 - [42] H. ZHANG AND S. SRA, *First-order methods for geodesically convex optimization*, in Proceedings of the 29th Annual Conference on Learning Theory, Proc. Mach. Learn. Res. 49 (2016), pp. 1617–1638; available at <http://proceedings.mlr.press/v49/>.
 - [43] S. ZHU, *The comparison geometry of Ricci curvature*, in Comparison Geometry, Math. Sci. Res. Inst. Publ. 30, Cambridge University Press, Cambridge, 1997, pp. 221–262.