IMA Journal of Numerical Analysis (2018) **00**, 1–33
doi:10.1093/imanum/drx080

Global rates of convergence for nonconvex optimization on manifolds

NICOLAS BOUMAL*
Mathematics Department and PACM, Princeton University, Princeton, NJ, USA
*Corresponding author: nboumal@math.princeton.edu

P.-A. ABSIL
ICTEAM Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium
absil@inma.ucl.ac.be

AND

CORALIA CARTIS
Mathematical Institute, University of Oxford, Oxford, UK
coralia.cartis@maths.ox.ac.uk

[Received on 15 March 2017; revised on 11 October 2017]

We consider the minimization of a cost function f on a manifold $\mathcal{M}$ using Riemannian gradient descent and Riemannian trust regions (RTR). We focus on satisfying necessary optimality conditions within a tolerance ε. Specifically, we show that, under Lipschitz-type assumptions on the pullbacks of f to the tangent spaces of $\mathcal{M}$, both of these algorithms produce points with Riemannian gradient smaller than ε in $\mathcal{O}(1/\varepsilon^2)$ iterations. Furthermore, RTR returns a point where also the Riemannian Hessian's least eigenvalue is larger than $-\varepsilon$ in $\mathcal{O}(1/\varepsilon^3)$ iterations. There are no assumptions on initialization. The rates match their (sharp) unconstrained counterparts as a function of the accuracy ε (up to constants) and hence are sharp in that sense. These are the first deterministic results for global rates of convergence to approximate first- and second-order Karush-Kuhn-Tucker points on manifolds. They apply in particular for optimization constrained to compact submanifolds of $\mathbb{R}^n$, under simpler assumptions.

Keywords: complexity; gradient descent; trust-region method; Riemannian optimization; optimization on manifolds.

1. Introduction

Optimization on manifolds is concerned with solving nonlinear and typically nonconvex computational problems of the form

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

where $\mathcal{M}$ is a (smooth) Riemannian manifold and $f\colon \mathcal{M} \to \mathbb{R}$ is a (sufficiently smooth) cost function (Gabay, 1982; Smith, 1994; Edelman *et al.*, 1998; Absil *et al.*, 2008). Applications abound in machine learning, computer vision, scientific computing, numerical linear algebra, signal processing, etc. In typical applications x is a matrix and $\mathcal{M}$ could be a Stiefel manifold of orthonormal frames (including spheres and groups of rotations), a Grassmann manifold of subspaces, a cone of positive definite matrices or simply a Euclidean space such as $\mathbb{R}^n$.

The standard theory for optimization on manifolds takes the standpoint that optimizing on a manifold $\mathcal{M}$ is not fundamentally different from optimizing in $\mathbb{R}^n$. Indeed, many classical algorithms from unconstrained nonlinear optimization such as gradient descent, nonlinear conjugate gradients, BFGS, Newton's method and trust-region methods (Ruszczyński, 2006; Nocedal & Wright, 1999) have been adapted to apply to the larger framework of (P) (Adler *et al.*, 2002; Absil *et al.*, 2007, 2008; Ring & Wirth, 2012; Huang *et al.*, 2015; Sato, 2016). Softwarewise, a few general toolboxes for optimization on manifolds exist now, for example, Manopt (Boumal *et al.*, 2014), PyManopt (Townsend *et al.*, 2016) and ROPTLIB (Huang *et al.*, 2016).

As (P) is typically nonconvex, one does not expect general purpose, efficient algorithms to converge to global optima of (P) in general. Indeed, the class of problems (P) includes known NP-hard problems. In fact, even computing *local* optima is NP-hard in general (Vavasis, 1991, Chapter 5). Nevertheless, one may still hope to compute points of $\mathcal{M}$ that satisfy first- and second-order necessary optimality conditions. These conditions take up the same form as in unconstrained nonlinear optimization, with *Riemannian* notions of gradient and Hessian. For $\mathcal{M}$ defined by equality constraints these conditions are equivalent to first- and second-order Karush-Kuhn-Tucker (KKT) conditions, but are simpler to manipulate because the Lagrangian multipliers are automatically determined.

The proposition below states these necessary optimality conditions. Recall that to each point x of $\mathcal{M}$ there corresponds a tangent space (a linearization) $\mathrm{T}_x\mathcal{M}$. The Riemannian gradient $\operatorname{grad} f(x)$ is the unique tangent vector at x such that $\mathrm{D}f(x)[\eta] = \langle \eta, \operatorname{grad} f(x)\rangle$ for all tangent vectors η, where $\langle \cdot, \cdot \rangle$ is the Riemannian metric on $\mathrm{T}_x\mathcal{M}$ and $\mathrm{D}f(x)[\eta]$ is the directional derivative of f at x along η. The Riemannian Hessian $\operatorname{Hess} f(x)$ is a symmetric operator on $\mathrm{T}_x\mathcal{M}$, corresponding to the derivative of the gradient vector field with respect to the Levi-Civita connection—see Absil *et al.* (2008, Chapter 5). These objects are easily computed in applications. A summary of relevant concepts about manifolds can be found in Appendix A.

PROPOSITION 1.1 (Necessary optimality conditions). Let $x \in \mathcal{M}$ be a local optimum for (P). If f is differentiable at x then $\operatorname{grad} f(x) = 0$. If f is twice differentiable at x then $\operatorname{Hess} f(x) \succeq 0$ (positive semidefinite).

Proof. See Yang *et al.* (2014, Rem. 4.2 and Cor. 4.2). □

A point $x \in \mathcal{M}$, which satisfies $\operatorname{grad} f(x) = 0$, is a *(first-order) critical point* (also called a stationary point). If x furthermore satisfies $\operatorname{Hess} f(x) \succeq 0$, it is a *second-order critical point*.

Existing theory for optimization algorithms on manifolds is mostly concerned with establishing global convergence to critical points without rates (where global means regardless of initialization), as well as local rates of convergence. For example, gradient descent is known to converge globally to critical points, and the convergence rate is linear once the iterates reach a *sufficiently small neighborhood* of the limit point (Absil *et al.*, 2008, Chapter 4). Early work of Udriste (1994) on local convergence rates even bounds distance to optimizers as a function of iteration count, assuming initialization in a set where the Hessian of f is positive definite, with lower and upper bounds on the eigenvalues; see also Absil *et al.* (2008, Thm. 4.5.6, Thm. 7.4.11). Such guarantees adequately describe the empirical behavior of those methods, but give no information about how many iterations are required to reach the local regime from an arbitrary initial point x_0; that is, the worst-case scenarios are not addressed.

For classical unconstrained nonlinear optimization this caveat has been addressed by bounding the number of iterations required by known algorithms to compute points that satisfy necessary optimality conditions within some tolerance, without assumptions on the initial iterate. Among others, Nesterov (2004) gives a proof that, for $\mathcal{M} = \mathbb{R}^n$ and Lipschitz differentiable f, gradient descent with an appropriate step size computes a point x where $\|\operatorname{grad} f(x)\| \leq \varepsilon$ in $\mathcal{O}\left(1/\varepsilon^2\right)$ iterations. This is sharp

(Cartis *et al.*, 2010). Cartis *et al.* (2012) prove the same for trust-region methods, and further show that if f is twice Lipschitz continuously differentiable, then a point x where $\|\operatorname{grad} f(x)\| \leq \varepsilon$ and $\operatorname{Hess} f(x) \succeq -\varepsilon \operatorname{Id}$ is computed in $\mathcal{O}\big(1/\varepsilon^3\big)$ iterations, also with examples showing sharpness.

In this paper we extend the unconstrained results to the larger class of optimization problems on manifolds (P). This work builds upon the original proofs (Nesterov, 2004; Cartis *et al.*, 2012) and on existing adaptations of gradient descent and trust-region methods to manifolds (Absil *et al.*, 2007, 2008). One key step is the identification of a set of relevant Lipschitz-type regularity assumptions, which allow the proofs to carry over from $\mathbb{R}^n$ to $\mathcal{M}$ with relative ease.

Main results

We state the main results here informally. We use the notion of *retraction* Retr_x (see Definition 2.1), which allows to map tangent vectors at x to points on $\mathcal{M}$. Iterates are related by $x_{k+1} = \operatorname{Retr}_{x_k}(\eta_k)$ for some tangent vector η_k at x_k (the step). Hence, $f \circ \operatorname{Retr}_x$ is a lift of the cost function from $\mathcal{M}$ to the tangent space at x. For $\mathcal{M} = \mathbb{R}^n$, the standard retraction gives $\operatorname{Retr}_{x_k}(\eta_k) = x_k + \eta_k$. By $\|\cdot\|$ we denote the norm associated to the Riemannian metric.

About gradient descent. (See Theorems 2.5 and 2.8.) For problem (P), if f is bounded below on $\mathcal{M}$ and $f \circ \operatorname{Retr}_x$ has Lipschitz gradient with constant L_g independent of x, then Riemannian gradient descent with constant step size $1/L_g$ or with backtracking Armijo line-search returns x with $\|\operatorname{grad} f(x)\| \leq \varepsilon$ in $\mathcal{O}\big(1/\varepsilon^2\big)$ iterations.

About trust regions. (See Theorem 3.4.) For problem (P), if f is bounded below on $\mathcal{M}$ and $f \circ \operatorname{Retr}_x$ has Lipschitz gradient with constant independent of x then Riemannian trust region (RTR) returns x with $\|\operatorname{grad} f(x)\| \leq \varepsilon_g$ in $\mathcal{O}\big(1/\varepsilon_g^2\big)$ iterations, under weak assumptions on the model quality. If furthermore $f \circ \operatorname{Retr}_x$ has Lipschitz Hessian with constant independent of x then RTR returns x with $\|\operatorname{grad} f(x)\| \leq \varepsilon_g$ and $\operatorname{Hess} f(x) \succeq -\varepsilon_H \operatorname{Id}$ in $\mathcal{O}\big(\max\big\{1/\varepsilon_H^3, 1/\varepsilon_g^2\varepsilon_H\big\}\big)$ iterations, provided the true Hessian is used in the model and a second-order retraction is used.

About compact submanifolds. (See Lemmas 2.4 and 3.1.) The first-order regularity conditions above hold in particular if $\mathcal{M}$ is a compact submanifold of a Euclidean space $\mathcal{E}$ (such as $\mathbb{R}^n$) and $f\colon \mathcal{E} \to \mathbb{R}$ has a locally Lipschitz continuous gradient. The second-order regularity conditions hold if furthermore f has a locally Lipschitz continuous Hessian on $\mathcal{E}$ and the retraction is second order (Definition 3.10).

Since the rates $\mathcal{O}\big(1/\varepsilon^2\big)$ and $\mathcal{O}\big(1/\varepsilon^3\big)$ are sharp for gradient descent and trust regions when $\mathcal{M} = \mathbb{R}^n$ (Cartis *et al.*, 2010, 2012), they are also sharp for $\mathcal{M}$ a generic Riemannian manifold. Below, constants are given explicitly, thus precisely bounding the total amount of work required in the worst case to attain a prescribed tolerance.

The theorems presented here are the first deterministic results about the worst-case iteration complexity of computing (approximate) first- and second-order critical points on manifolds. The choice of analysing Riemannian gradient descent and RTR first is guided by practical concerns, as these are among the most commonly used methods on manifolds so far.

The proposed complexity bounds are particularly relevant when applied to problems for which second-order necessary optimality conditions are also sufficient. See for example Sun *et al.* (2015, 2016), Boumal (2015b, 2016), Bandeira *et al.* (2016), Bhojanapalli *et al.* (2016), Ge *et al.* (2016) and the example in Section 4.

Related work

The complexity of Riemannian optimization is discussed in a few recent lines of work. Zhang & Sra (2016) treat geodesically convex problems over Hadamard manifolds. This is a remarkable extension of important pieces of classical convex optimization theory to manifolds with negative curvature. Because of the focus on geodesically convex problems, those results do not apply to the more general problem (P), but have the clear advantage of guaranteeing global optimality. In Zhang *et al.* (2016), which appeared a day before the present paper on public repositories, the authors also study the iteration complexity of nonconvex optimization on manifolds. Their results differ from the ones presented here in that they focus on *stochastic* optimization algorithms, aiming for first-order conditions. Their results assume bounded curvature for the manifold. Furthermore, their analysis relies on the Riemannian exponential map, whereas we cover the more general class of retraction maps (which is computationally advantageous). We also do not use the notions of Riemannian parallel transport or logarithmic map, which, in our view, makes for a simpler analysis.

Sun *et al.* (2015, 2016) consider dictionary learning and phase retrieval and show that these problems, when appropriately framed as optimization on a manifold, are low-dimensional and have no spurious local optimizers. They derive the complexity of RTR specialized to their application. In particular, they combine the global rate with a local convergence rate, which allows them to establish an overall better complexity than $\mathcal{O}\left(1/\varepsilon^3\right)$, but with an idealized version of the algorithm and restricted to these relevant applications. In this paper we favor a more general approach, focused on algorithms closer to the ones implemented in practice.

Recent work by Bento *et al.* (2017) (which appeared after a first version of this paper) focuses on iteration complexity of gradient, subgradient and proximal point methods for the case of convex cost functions on manifolds, using the exponential map as retraction.

For the classical, unconstrained case, optimal complexity bounds of order $\mathcal{O}\left(1/\varepsilon^{1.5}\right)$ to generate x with $\|\operatorname{grad} f(x)\| \leq \varepsilon$ have also been given for cubic regularization methods (Cartis *et al.*, 2011a, b) and sophisticated trust region variants (Curtis *et al.*, 2016). Bounds for regularization methods can be further improved if higher-order derivatives are available (Birgin *et al.*, 2017).

Worst-case evaluation complexity bounds have been extended to constrained smooth problems in Cartis *et al.* (2014, 2015a,b). There, it is shown that some carefully devised, albeit impractical, phase 1–phase 2 methods can compute approximate KKT points with global rates of convergence of the same order as in the unconstrained case. We note that when the constraints are convex (but the objective may not be), practical, feasible methods have been devised (Cartis *et al.*, 2015a) that connect to our approach below. Second-order optimality for the case of convex constraints with nonconvex cost has been recently addressed in Cartis *et al.* (2016).

2. Riemannian gradient descent methods

Consider the generic Riemannian descent method described in Algorithm 1. We first prove that, provided sufficient decrease in the cost function is achieved at each iteration, the algorithm computes a point x_k such that $\|\operatorname{grad} f(x_k)\| \leq \varepsilon$ with $k = \mathcal{O}\left(1/\varepsilon^2\right)$. Then we propose a Lipschitz-type assumption, which is sufficient to guarantee that simple strategies to pick the steps η_k indeed ensure sufficient decrease. The proofs parallel the standard ones (Nesterov, 2004, Sect. 1.2.3). The main novelty is the careful extension to the Riemannian setting, which requires the well-known notion of retraction (Definition 2.1) and the new Assumption 2.6 (see below).

The step η_k is a tangent vector to $\mathcal{M}$ at x_k. Because $\mathcal{M}$ is nonlinear (in general) the operation $x_k + \eta_k$ is undefined. The notion of *retraction* provides a theoretically sound replacement. Informally, $x_{k+1} = \mathrm{Retr}_{x_k}(\eta_k)$ is *a point on $\mathcal{M}$ that one reaches by moving away from x_k, along the direction η_k, while remaining on the manifold*. The Riemannian exponential map (which generates geodesics) is a retraction. The crucial point is that many other maps are retractions, often far less difficult to compute than the exponential. The definition of retraction below can be traced back to Shub (1986) and it appears under that name in Adler *et al.* (2002); see also Absil *et al.* (2008, Def. 4.1.1 and Sect. 4.10) for additional references.

DEFINITION 2.1 (Retraction). A *retraction* on a manifold $\mathcal{M}$ is a smooth mapping Retr from the tangent bundle[1] $\mathrm{T}\mathcal{M}$ to $\mathcal{M}$ with the following properties. Let $\mathrm{Retr}_x\colon \mathrm{T}_x\mathcal{M} \to \mathcal{M}$ denote the restriction of Retr to $\mathrm{T}_x\mathcal{M}$:

(i) $\mathrm{Retr}_x(0_x) = x$, where 0_x is the zero vector in $\mathrm{T}_x\mathcal{M}$;

(ii) the differential of Retr_x at 0_x, $\mathrm{D}\,\mathrm{Retr}_x(0_x)$, is the identity map.

These combined conditions ensure retraction curves $t \mapsto \mathrm{Retr}_x(t\eta)$ agree up to first order with geodesics passing through x with velocity η, around $t = 0$. Sometimes we allow Retr_x to be defined only locally, in a closed ball of radius $\varrho(x) > 0$ centered at 0_x in $\mathrm{T}_x\mathcal{M}$.

In linear spaces such as $\mathbb{R}^n$ the typical choice is $\mathrm{Retr}_x(\eta) = x + \eta$. On the sphere, a popular choice is $\mathrm{Retr}_x(\eta) = \frac{x+\eta}{\|x+\eta\|}$.

REMARK 2.2 If the retraction at x_k is defined only in a ball of radius $\varrho_k = \varrho(x_k)$ around the origin in $\mathrm{T}_{x_k}\mathcal{M}$, we limit the size of step η_k to ϱ_k. Theorems in this section provide a complexity result provided $\varrho = \inf_k \varrho_k > 0$. If the *injectivity radius* of the manifold is positive, retractions satisfying the condition $\inf_{x\in\mathcal{M}} \varrho(x) > 0$ exist. In particular, compact manifolds have positive injectivity radius (Chavel, 2006, Thm. III.2.3). The option to limit the step sizes is also useful when the constant L_g in Assumption 2.6 below does not exist globally.

The two central assumptions and a general theorem about Algorithm 1 follow.

ASSUMPTION 2.3 (Lower bound). There exists $f^* > -\infty$ such that $f(x) \geq f^*$ for all $x \in \mathcal{M}$.

ASSUMPTION 2.4 (Sufficient decrease). There exist $c, c' > 0$ such that, for all $k \geq 0$,

$$f(x_k) - f(x_{k+1}) \geq \min\left(c\|\mathrm{grad}\, f(x_k)\|, c'\right)\|\mathrm{grad}\, f(x_k)\|.$$

Algorithm 1 Generic Riemannian descent algorithm

Given: $f\colon \mathcal{M} \to \mathbb{R}$ differentiable, a retraction Retr on $\mathcal{M}$, $x_0 \in \mathcal{M}$, $\varepsilon > 0$
Init: $k \leftarrow 0$
while $\|\mathrm{grad}\, f(x_k)\| > \varepsilon$ **do**
  Pick $\eta_k \in \mathrm{T}_{x_k}\mathcal{M}$ (e.g., as in Theorem 2.8 or Theorem 2.11)
  $x_{k+1} = \mathrm{Retr}_{x_k}(\eta_k)$
  $k \leftarrow k + 1$
end while
return x_k ▷ $\|\mathrm{grad}\, f(x_k)\| \leq \varepsilon$

[1] Informally, the tangent bundle $\mathrm{T}\mathcal{M}$ is the set of all pairs (x, η_x) where $x \in \mathcal{M}$ and $\eta_x \in \mathrm{T}_x\mathcal{M}$. See the reference for a proper definition of $\mathrm{T}\mathcal{M}$ and of what it means for Retr to be smooth.

THEOREM 2.5 Under Assumptions 2.3 and 2.4, Algorithm 1 returns $x \in \mathcal{M}$ satisfying $f(x) \leq f(x_0)$ and $\|\operatorname{grad} f(x)\| \leq \varepsilon$ in at most

$$\left\lceil \frac{f(x_0) - f^*}{c} \cdot \frac{1}{\varepsilon^2} \right\rceil$$

iterations, provided $\varepsilon \leq \frac{c'}{c}$. If $\varepsilon > \frac{c'}{c}$, at most $\left\lceil \frac{f(x_0)-f^*}{c'} \cdot \frac{1}{\varepsilon} \right\rceil$ iterations are required.

Proof. If Algorithm 1 executes $K-1$ iterations without terminating, then $\|\operatorname{grad} f(x_k)\| > \varepsilon$ for all k in $0, \ldots, K-1$. Then using Assumptions 2.3 and 2.4 in a classic telescoping sum argument gives

$$f(x_0) - f^* \geq f(x_0) - f(x_K) = \sum_{k=0}^{K-1} f(x_k) - f(x_{k+1}) > K \min(c\varepsilon, c')\varepsilon.$$

By contradiction, the algorithm must have terminated if $K \geq \frac{f(x_0)-f^*}{\min(c\varepsilon,c')\varepsilon}$. □

To ensure Assumption 2.4 with simple rules for the choice of η_k it is necessary to restrict the class of functions f. For the particular case $\mathcal{M} = \mathbb{R}^n$ and $\operatorname{Retr}_x(\eta) = x + \eta$, the classical tion is to require f to have a Lipschitz continuous gradient (Nesterov, 2004), that is, existence of L_g such that

$$\forall\, x, y \in \mathbb{R}^n, \quad \|\operatorname{grad} f(x) - \operatorname{grad} f(y)\| \leq L_g \|x - y\|. \tag{2.1}$$

As we argue momentarily, generalizing this property to manifolds is impractical. On the other hand, it is well known that (2.1) implies (see for example Nesterov, 2004, Lemma 1.2.3; see also Berger, 2017, Appendix A for a converse):

$$\forall\, x, y \in \mathbb{R}^n, \quad \left| f(y) - \left[f(x) + \langle y - x, \operatorname{grad} f(x) \rangle \right] \right| \leq \frac{L_g}{2} \|y - x\|^2. \tag{2.2}$$

It is the latter we adapt to manifolds. Consider the *pullback*[2] $\hat{f}_x = f \circ \operatorname{Retr}_x \colon \mathrm{T}_x\mathcal{M} \to \mathbb{R}$, conveniently defined on a vector space. It follows from the definition of retraction that $\operatorname{grad} \hat{f}_x(0_x) = \operatorname{grad} f(x)$.[3] Thinking of x as x_k and of y as $\operatorname{Retr}_{x_k}(\eta)$, we require the following.

ASSUMPTION 2.6 (Restricted Lipschitz-type gradient for pullbacks). There exists $L_g \geq 0$ such that, for all x_k among $x_0, x_1, \ldots$ generated by a specified algorithm, the composition $\hat{f}_k = f \circ \operatorname{Retr}_{x_k}$ satisfies

$$\left| \hat{f}_k(\eta) - \left[f(x_k) + \langle \eta, \operatorname{grad} f(x_k) \rangle \right] \right| \leq \frac{L_g}{2} \|\eta\|^2 \tag{2.3}$$

for all $\eta \in \mathrm{T}_{x_k}\mathcal{M}$ such that $\|\eta\| \leq \varrho_k$.[4] In words, the pullbacks $\hat{f}_k$, possibly restricted to certain balls, are uniformly well approximated by their first-order Taylor expansions around the origin.

---

[2] The composition $f \circ \operatorname{Retr}_x$ is called the pullback because it, quite literally, pulls back the cost function f from the manifold $\mathcal{M}$ to the linear space $\mathrm{T}_x\mathcal{M}$.

[3] $\forall\, \eta \in \mathrm{T}_x\mathcal{M}, \langle \operatorname{grad} \hat{f}_x(0_x), \eta \rangle = \mathrm{D}\hat{f}_x(0_x)[\eta] = \mathrm{D}f(x)[\mathrm{D}\operatorname{Retr}_x(0_x)[\eta]] = \mathrm{D}f(x)[\eta] = \langle \operatorname{grad} f(x), \eta \rangle$.

[4] See Remark 2.2; $\rho_k = \infty$ is valid if the retraction is globally defined and f is sufficiently nice (for example, Lemma 2.7).

To the best of our knowledge, this specific tion has not been used to analyse convergence of optimization algorithms on manifolds before. As will become clear, it allows for simple extensions of existing proofs in $\mathbb{R}^n$.

Notice that if each $\hat{f}_k$ has a Lipschitz continuous gradient with constant L_g independent of k,[5] then Assumption 2.6 holds but the reverse is not necessarily true as Assumption 2.6 gives a special role to the origin. In this sense the condition on $\hat{f}_k$ is weaker than Lipschitz continuity of the gradient of $\hat{f}_k$. On the other hand, we are requiring this condition to hold for all x_k with the same constant L_g. This is why we call the condition *Lipschitz type* rather than Lipschitz.

The following lemma states that if $\mathcal{M}$ is a compact submanifold of $\mathbb{R}^n$ then a sufficient condition for Assumption 2.6 to hold is for $f\colon \mathbb{R}^n \to \mathbb{R}$ to have locally Lipschitz continuous gradient (so that it has Lipschitz continuous gradient on any compact subset of $\mathbb{R}^n$). The proof is in Appendix B.

LEMMA 2.7 Let $\mathcal{E}$ be a Euclidean space (for example, $\mathcal{E} = \mathbb{R}^n$) and let $\mathcal{M}$ be a compact Riemannian submanifold of $\mathcal{E}$. Let Retr be a retraction on $\mathcal{M}$ (globally[6] defined). If $f\colon \mathcal{E} \to \mathbb{R}$ has Lipschitz continuous gradient in the convex hull of $\mathcal{M}$, then the pullbacks $f \circ \mathrm{Retr}_x$ satisfy (2.3) globally with some constant L_g independent of x; hence, Assumption 2.6 holds for any sequence of iterates and with $\varrho_k = \infty$ for all k.

There are mainly two difficulties with generalizing (2.1) directly to manifolds. First, $\mathrm{grad}\, f(x)$ and $\mathrm{grad}\, f(y)$ live in two different tangent spaces, so that their difference is not defined; instead, $\mathrm{grad}\, f(x)$ must be *transported* to $\mathrm{T}_y\mathcal{M}$, which requires the introduction of a *parallel transport* $\mathrm{P}_{x\to y}\colon \mathrm{T}_x\mathcal{M} \to \mathrm{T}_y\mathcal{M}$ along a minimal geodesic connecting x and y. Second, the right-hand side $\|x - y\|$ should become $\mathrm{dist}(x, y)$: the *geodesic distance* on $\mathcal{M}$. Both notions involve subtle definitions and transports may not be defined on all of $\mathcal{M}$. Overall, the resulting tion would read that there exists L_g such that

$$\forall\, x, y \in \mathcal{M}, \quad \|\mathrm{P}_{x\to y}\mathrm{grad}\, f(x) - \mathrm{grad}\, f(y)\| \leq L_g \mathrm{dist}(x, y). \tag{2.4}$$

It is of course possible to work with (2.4)—see for example Absil *et al.* (2008, Def. 7.4.3) and recent work of Zhang & Sra (2016) and Zhang *et al.* (2016)—but we argue that it is conceptually and computationally advantageous to avoid it if possible. The computational advantage comes from the freedom in Assumption 2.6 to work with any retraction, whereas parallel transport and geodesic distance are tied to the exponential map.

We note that, if the retraction is the exponential map, then it is known that Assumption 2.6 holds if (2.4) holds—see for example Bento *et al.* (2017, Def. 2.2 and Lemma 2.1).

2.1 *Fixed step-size gradient descent method*

Leveraging the regularity Assumption 2.6, an easy strategy is to pick the step η_k as a fixed scaling of the negative gradient, possibly restricted to a ball of radius ϱ_k.

---

[5] This holds in particular in the classical setting $\mathcal{M} = \mathbb{R}^n$, $\mathrm{Retr}_x(\eta) = x + \eta$ and $\mathrm{grad}\, f$ is L_g-Lipschitz.

[6] This is typically not an issue in practice. For example, globally defined, practical retractions are known for the sphere, Stiefel manifold, orthogonal group, their products and many others (Absil *et al.*, 2008, Chapter 4).

THEOREM 2.8 (Riemannian gradient descent with fixed step size). Under Assumptions 2.3 and 2.6, Algorithm 1 with the explicit strategy

$$\eta_k = -\min\left(\frac{1}{L_g}, \frac{\varrho_k}{\|\operatorname{grad} f(x_k)\|}\right)\operatorname{grad} f(x_k)$$

returns a point $x \in \mathcal{M}$ satisfying $f(x) \leq f(x_0)$ and $\|\operatorname{grad} f(x)\| \leq \varepsilon$ in at most

$$\left\lceil 2\left(f(x_0) - f^*\right) L_g \cdot \frac{1}{\varepsilon^2} \right\rceil$$

iterations provided $\varepsilon \leq \varrho L_g$, where $\varrho = \inf_k \rho_k$. If $\varepsilon > \varrho L_g$ the algorithm succeeds in at most $\left\lceil 2\left(f(x_0) - f^*\right) \frac{1}{\varrho} \cdot \frac{1}{\varepsilon} \right\rceil$ iterations. Each iteration requires one cost and gradient evaluation and one retraction.

Proof. The regularity Assumption 2.6 provides an upper bound for the pullback for all k:

$$\forall\, \eta \in \mathrm{T}_{x_k}\mathcal{M} \text{ with } \|\eta\| \leq \varrho_k, \quad f(\operatorname{Retr}_{x_k}(\eta)) \leq f(x_k) + \langle \eta, \operatorname{grad} f(x_k)\rangle + \frac{L_g}{2}\|\eta\|^2. \tag{2.5}$$

For the given choice of η_k and using $x_{k+1} = \operatorname{Retr}_{x_k}(\eta_k)$, it follows easily that

$$f(x_k) - f(x_{k+1}) \geq \min\left(\frac{\|\operatorname{grad} f(x_k)\|}{L_g}, \varrho_k\right)\left[1 - \frac{L_g}{2}\min\left(\frac{1}{L_g}, \frac{\varrho_k}{\|\operatorname{grad} f(x_k)\|}\right)\right]\|\operatorname{grad} f(x_k)\|.$$

The term in brackets is at least 1/2. Thus, Assumption 2.4 holds with $c = \frac{1}{2L_g}$ and $c' = \frac{\varrho}{2}$, allowing us to conclude with Theorem 2.3. □

COROLLARY 2.9 If there are no step-size restrictions in Theorem 2.5 ($\rho_k \equiv \infty$), the explicit strategy

$$\eta_k = -\frac{1}{L_g}\operatorname{grad} f(x_k)$$

returns a point $x \in \mathcal{M}$ satisfying $f(x) \leq f(x_0)$ and $\|\operatorname{grad} f(x)\| \leq \varepsilon$ in at most

$$\left\lceil 2\left(f(x_0) - f^*\right) L_g \cdot \frac{1}{\varepsilon^2} \right\rceil$$

iterations for any $\varepsilon > 0$.

2.2 *Gradient descent with backtracking Armijo line-search*

The following lemma shows that a basic Armijo-type backtracking line-search, Algorithm 2, computes a step η_k satisfying Assumption 2.4 in a bounded number of function calls, without the need to know L_g. The statement allows search directions other than $-\operatorname{grad} f(x_k)$, provided they remain 'related' to $-\operatorname{grad} f(x_k)$. This result is well known in the Euclidean case and carries over seamlessly under Assumption 2.6.

Algorithm 2 Backtracking Armijo line-search

Given: $x_k \in \mathcal{M}$, $\eta_k^0 \in \mathrm{T}_{x_k}\mathcal{M}$, $\bar{t}_k > 0$, $c_1 \in (0,1)$, $\tau \in (0,1)$
Init: $t \leftarrow \bar{t}_k$
while $f(x_k) - f\left(\mathrm{Retr}_{x_k}\left(t \cdot \eta_k^0\right)\right) < c_1 t \left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle$ **do**
    $t \leftarrow \tau \cdot t$
end while
return t and $\eta_k = t\eta_k^0$

LEMMA 2.10 For each iteration k of Algorithm 1, let $\eta_k^0 \in \mathrm{T}_{x_k}\mathcal{M}$ be the initial search direction to be considered for line-search. Assume there exist constants $c_2 \in (0, 1]$ and $0 < c_3 \leq c_4$ such that, for all k,

$$\left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle \geq c_2 \|\mathrm{grad}\, f(x_k)\| \left\|\eta_k^0\right\| \qquad \text{and} \qquad c_3 \|\mathrm{grad}\, f(x_k)\| \leq \left\|\eta_k^0\right\| \leq c_4 \|\mathrm{grad}\, f(x_k)\|.$$

Under Assumption 2.6, backtracking Armijo (Algorithm 2) with initial step size $\bar{t}_k$ such that $\bar{t}_k \left\|\eta_k^0\right\| \leq \varrho_k$ returns a positive t and $\eta_k = t\eta_k^0$ such that

$$f(x_k) - f\left(\mathrm{Retr}_{x_k}(\eta_k)\right) \geq c_1 c_2 c_3 t \|\mathrm{grad}\, f(x_k)\|^2 \qquad \text{and} \qquad t \geq \min\left(\bar{t}_k, \frac{2\tau c_2 (1 - c_1)}{c_4 L_g}\right) \tag{2.6}$$

in

$$1 + \log_\tau\left(t/\bar{t}_k\right) \leq \max\left(1, 2 + \left\lceil \log_{\tau^{-1}}\left(\frac{c_4 \bar{t}_k L_g}{2c_2(1 - c_1)}\right)\right\rceil\right)$$

retractions and cost evaluations (not counting evaluation of f at x_k).

Proof. See Appendix C. □

The previous discussion can be particularized to bound the amount of work required by a gradient descent method using a backtracking Armijo line-search on manifolds. The constant L_g appears in the bounds but needs not be known. Note that, at iteration k, the last cost evaluation of the line-search algorithm is the cost at x_{k+1}: it need not be recomputed.

THEOREM 2.11 (Riemannian gradient descent with backtracking line-search). Under Assumptions 2.3 and 2.6, Algorithm 1 with Algorithm 2 for line-search using initial search direction $\eta_k^0 = -\mathrm{grad}\, f(x_k)$ with parameters c_1, τ and $\bar{t}_k \triangleq \min\left(\bar{t}, \varrho_k / \|\mathrm{grad}\, f(x_k)\|\right)$ for some $\bar{t} > 0$ returns a point $x \in \mathcal{M}$ satisfying $f(x) \leq f(x_0)$ and $\|\mathrm{grad}\, f(x)\| \leq \varepsilon$ in at most

$$\left\lceil \frac{f(x_0) - f^*}{c_1 \min\left(\bar{t}, \frac{2\tau(1-c_1)}{L_g}\right)} \cdot \frac{1}{\varepsilon^2} \right\rceil$$

iterations, provided $\varepsilon \leq \frac{\varrho}{\min\left(\bar{t}, \frac{2\tau(1-c_1)}{L_g}\right)} \triangleq c$, where $\varrho = \inf_k \varrho_k$. If $\varepsilon > c$, the algorithm succeeds in at most $\left\lceil \frac{f(x_0)-f^*}{c_1 \varrho} \cdot \frac{1}{\varepsilon} \right\rceil$ iterations. After computing $f(x_0)$ and $\operatorname{grad} f(x_0)$, each iteration requires one gradient evaluation and at most $\max\left(1, 2 + \left\lceil \log_{\tau^{-1}}\left(\frac{\bar{t} L_g}{2(1-c_1)}\right)\right\rceil\right)$ cost evaluations and retractions.

Proof. Using $\eta_k^0 = -\operatorname{grad} f(x_k)$, one can take $c_2 = c_3 = c_4 = 1$ in Lemma 2.7. Equation (2.6) in that lemma combined with the definition of $\bar{t}_k$ ensures

$$f(x_k) - f(x_{k+1}) \geq c_1 \min\left(\bar{t}, \frac{2\tau(1-c_1)}{L_g}, \frac{\varrho_k}{\|\operatorname{grad} f(x_k)\|}\right) \|\operatorname{grad} f(x_k)\|^2.$$

Thus, Assumption 2.4 holds with $c = c_1 \min\left(\bar{t}, \frac{2\tau(1-c_1)}{L_g}\right)$ and $c' = c_1 \varrho$. Conclude with Theorem 2.3.□

3. RTR methods

The RTR method is a generalization of the classical trust-region method to manifolds (Absil *et al.*, 2007; Conn *et al.*, 2000)—see Algorithm 3. The algorithm is initialized with a point $x_0 \in \mathcal{M}$ and a trust-region radius Δ_0. At iteration k, the pullback $\hat{f}_k = f \circ \operatorname{Retr}_{x_k}$ is approximated by a model $\hat{m}_k \colon \mathrm{T}_{x_k}\mathcal{M} \to \mathbb{R}$,

$$\hat{m}_k(\eta) = f(x_k) + \langle \eta, \operatorname{grad} f(x_k) \rangle + \tfrac{1}{2} \langle \eta, H_k[\eta] \rangle, \tag{3.1}$$

where $H_k \colon \mathrm{T}_{x_k}\mathcal{M} \to \mathrm{T}_{x_k}\mathcal{M}$ is a map chosen by the user. The tentative step η_k is obtained by approximately solving the associated trust-region subproblem:

$$\min_{\eta \in \mathrm{T}_{x_k}\mathcal{M}} \hat{m}_k(\eta) \quad \text{subject to} \quad \|\eta\| \leq \Delta_k. \tag{3.2}$$

The candidate next iterate $x_k^+ = \operatorname{Retr}_{x_k}(\eta_k)$ is accepted ($x_{k+1} = x_k^+$) if the actual cost decrease $f(x_k) - f\left(x_k^+\right)$ is a sufficiently large fraction of the model decrease $\hat{m}_k\left(0_{x_k}\right) - \hat{m}_k(\eta_k)$. Otherwise, the candidate is rejected ($x_{k+1} = x_k$). Depending on the level of agreement between the model decrease and actual decrease, the trust-region radius Δ_k can be reduced, kept unchanged or increased, but never above some parameter $\bar{\Delta}$. The parameter $\bar{\Delta}$ can be used in particular in case of a nonglobally defined retraction or if the regularity conditions on the pullbacks hold only locally.

We establish worst-case iteration complexity bounds for the computation of points $x \in \mathcal{M}$ such that $\|\operatorname{grad} f(x)\| \leq \varepsilon_g$ and $\operatorname{Hess} f(x) \succcurlyeq -\varepsilon_H \operatorname{Id}$, where $\operatorname{Hess} f(x)$ is the Riemannian Hessian of f at x. Besides Lipschitz-type conditions on the problem itself, essential algorithmic requirements are that (i) the models $\hat{m}_k$ should agree sufficiently with the pullbacks $\hat{f}_k$ (locally) and (ii) sufficient decrease in the model should be achieved at each iteration. The analysis presented here is a generalization of the one in Cartis *et al.* (2012) to manifolds.

3.1 *Regularity conditions*

In what follows, for iteration k, we make assumptions involving the ball of radius $\Delta_k \leq \bar{\Delta}$ around 0_{x_k} in the tangent space at x_k. If Retr_x is defined only in a ball of radius $\varrho(x)$, one (conservative) strategy to ensure $\varrho_k \geq \Delta_k$ as required in the assumption below is to set $\bar{\Delta} \leq \inf_{x \in \mathcal{M}: f(x) \leq f(x_0)} \varrho(x)$, provided this is positive (see Remark 2.2).

ASSUMPTION 3.1 (Restricted Lipschitz-type gradient for pullbacks). Assumption 2.6 holds in the respective trust regions of the iterates produced by Algorithm 3, that is, with $\varrho_k \geq \Delta_k$.

Algorithm 3 RTR, modified to attain second-order optimality

Parameters: $\bar{\Delta} > 0$, $0 < \rho' < 1/4$, $\varepsilon_g > 0$, $\varepsilon_H > 0$
Input: $x_0 \in \mathcal{M}$, $0 < \Delta_0 \leq \bar{\Delta}$
Init: $k \leftarrow 0$
while true do
  if $\|\operatorname{grad} f(x_k)\| > \varepsilon_g$ **then** ▷ First-order step
    Obtain $\eta_k \in \mathrm{T}_{x_k}\mathcal{M}$ satisfying Assumption 3.6 (e.g., Lemma 3.7)
  else if $\varepsilon_H < \infty$ **then** ▷ Second-order step
    if $\lambda_{\min}(H_k) < -\varepsilon_H$ **then**
      Obtain $\eta_k \in \mathrm{T}_{x_k}\mathcal{M}$ satisfying Assumption 3.8 (e.g., Lemma 3.9)
    else
      return x_k ▷ $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$ and $\lambda_{\min}(H_k) \geq -\varepsilon_H$
    end if
  else
    return x_k ▷ $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$
  end if
  Compute

$$\rho_k = \frac{\hat{f}_k(0_{x_k}) - \hat{f}_k(\eta_k)}{\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k)} \tag{3.3}$$

  $$\Delta_{k+1} = \begin{cases} \frac{1}{4}\Delta_k & \text{if } \rho_k < \frac{1}{4} \text{ (poor model-cost agreement)}, \\ \min\left(2\Delta_k, \bar{\Delta}\right) & \text{if } \rho_k > \frac{3}{4} \text{ and } |\eta_k\| = \Delta_k \text{ (good agreement, limiting TR)}, \\ \Delta_k & \text{otherwise.} \end{cases}$$

  $$x_{k+1} = \begin{cases} \operatorname{Retr}_{x_k}(\eta_k) & \text{if } \rho_k > \rho' \text{ (accept the step)}, \\ x_k & \text{otherwise (reject).} \end{cases}$$

  $k \leftarrow k + 1$
end while

ASSUMPTION 3.2 (Restricted Lipschitz-type Hessian for pullbacks). If $\varepsilon_H < \infty$ there exists $L_H \geq 0$ such that, for all x_k among $x_0, x_1, \ldots$ generated by Algorithm 3 and such that $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$, $\hat{f}_k$ satisfies

$$\left|\hat{f}_k(\eta) - \left[f(x_k) + \langle \eta, \operatorname{grad} f(x_k)\rangle + \frac{1}{2}\left\langle \eta, \nabla^2 \hat{f}_k(0_{x_k})[\eta]\right\rangle\right]\right| \leq \frac{L_H}{6}\|\eta\|^3 \tag{3.4}$$

for all $\eta \in \mathrm{T}_{x_k}\mathcal{M}$ such that $\|\eta\| \leq \Delta_k$.

As discussed in Section 3.5 below, if Retr is a *second-order* retraction, then $\nabla^2 \hat{f}_k\left(0_{x_k}\right)$ coincides with the Riemannian Hessian of f at x_k.

In the previous section, Lemma 2.7 gives a sufficient condition for Assumption 3.1 to hold; we complement this statement with a sufficient condition for Assumption 3.2 to hold as well. In a nutshell, if $\mathcal{M}$ is a compact submanifold of $\mathbb{R}^n$ and $f\colon \mathbb{R}^n \to \mathbb{R}$ has locally Lipschitz continuous Hessian, then both assumptions hold.

LEMMA 3.3 Let $\mathcal{E}$ be a Euclidean space (for example, $\mathcal{E} = \mathbb{R}^n$) and let $\mathcal{M}$ be a compact Riemannian submanifold of $\mathcal{E}$. Let Retr be a second-order retraction on $\mathcal{M}$ (globally defined). If $f\colon \mathcal{E} \to \mathbb{R}$ has Lipschitz continuous Hessian in the convex hull of $\mathcal{M}$, then the pullbacks $f \circ \mathrm{Retr}_x$ obey (3.4) with some constant L_H independent of x; hence, Assumption 3.2 holds for any sequence of iterates and trust-region radii.

The proof is in Appendix B. Here too, if $\mathcal{M}$ is a Euclidean space and $\mathrm{Retr}_x(\eta) = x + \eta$, then Assumptions 3.1 and 3.2 are satisfied if f has Lipschitz continuous Hessian in the usual sense.

3.2 *Assumptions about the models*

The model at iteration k is the function $\hat{m}_k$ (3.1) whose purpose is to approximate the pullback $\hat{f}_k = f \circ \mathrm{Retr}_{x_k}$. It involves a map $H_k\colon \mathrm{T}_{x_k}\mathcal{M} \to \mathrm{T}_{x_k}\mathcal{M}$. Depending on the type of step being performed (aiming for first- or second-order optimality conditions), we require different properties of the maps H_k. Conditions for first-order optimality are particularly lax.

ASSUMPTION 3.4 If $\|\mathrm{grad}\, f(x_k)\| > \varepsilon_g$ (so that we are aiming only for a first-order condition at this step) then H_k is radially linear. That is,

$$\forall\, \eta \in \mathrm{T}_{x_k}\mathcal{M}, \forall\, \alpha \geq 0, \quad H_k[\alpha\eta] = \alpha H_k[\eta]. \tag{3.5}$$

Furthermore, there exists $c_0 \geq 0$ (the same for all first-order steps) such that

$$\|H_k\| \triangleq \sup_{\eta \in \mathrm{T}_{x_k}\mathcal{M}:\|\eta\|\leq 1} \langle \eta, H_k[\eta]\rangle \leq c_0. \tag{3.6}$$

Radial linearity and boundedness are sufficient to ensure first-order agreement between $\hat{m}_k$ and $\hat{f}_k$. This relaxation from complete linearity of H_k, which would be the standard assumption, notably allows the use of nonlinear finite difference approximations of the Hessian (Boumal, 2015a). To reach second-order agreement, the conditions are stronger.

ASSUMPTION 3.5 If $\|\mathrm{grad}\, f(x_k)\| \leq \varepsilon_g$ and $\varepsilon_H < \infty$ (so that we are aiming for a second-order condition), then H_k is *linear and symmetric*. Furthermore, H_k is close to $\nabla^2 \hat{f}_k\left(0_{x_k}\right)$ along η_k in the sense that there exists $c_1 \geq 0$ (the same for all second-order steps) such that

$$\left|\left\langle \eta_k, \left(\nabla^2 \hat{f}_k(0_{x_k}) - H_k\right)[\eta_k]\right\rangle\right| \leq \frac{c_1 \Delta_k}{3}\|\eta_k\|^2. \tag{3.7}$$

The smaller Δ_k, the more precisely H_k must approximate the Hessian of the pullback along η_k. Lemma 3.6 (below) shows Δ_k is lower bounded in relation with ε_g and ε_H.

Equation (3.7) involves η_k, the ultimately chosen step that typically depends on H_k. The stronger condition below does not reference η_k and yet still ensures (3.7) is satisfied:

$$\left\| \nabla^2 \hat{f}_k \left(0_{x_k}\right) - H_k \right\| \leq \frac{c_1 \Delta_k}{3}.$$

Refer to Section 3.5 to relate H_k, $\nabla^2 \hat{f}_k \left(0_{x_k}\right)$ and $\operatorname{Hess} f(x_k)$.

3.3 *Assumptions about sufficient model decrease*

The steps η_k can be obtained in a number of ways, leading to different local convergence rates and empirical performance. As far as global convergence guarantees are concerned though, the requirements are modest. It is required only that, at each iteration, the candidate η_k induces sufficient decrease *in the model*. Known explicit strategies achieve these decreases. In particular, solving the trust-region subproblem (3.2) within some tolerance (which can be done in polynomial time if H_k is linear; see Vavasis, 1991, Sect. 4.3) is certain to satisfy the assumptions. The Steihaug–Toint truncated conjugate gradients method is a popular choice (Toint, 1981; Steihaug, 1983; Conn *et al.*, 2000; Absil *et al.*, 2007). See also Sorensen (1982) and Moré & Sorensen (1983) for more about the trust-region subproblem. Here we describe simpler yet satisfactory strategies. For first-order steps we require the following.

Assumption 3.6 There exists $c_2 > 0$ such that, for all k such that $\|\operatorname{grad} f(x_k)\| > \varepsilon_g$, the step η_k satisfies

$$\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k) \geq c_2 \min\left(\Delta_k, \frac{\varepsilon_g}{c_0}\right) \varepsilon_g. \tag{3.8}$$

As is well known, the explicitly computable *Cauchy step* satisfies this requirement. For convenience let $g_k = \operatorname{grad} f(x_k)$. By definition the Cauchy step minimizes $\hat{m}_k$ (3.1) in the trust region along the steepest descent direction $-g_k$. Owing to radial linearity (Assumption 3.4), this reads

$$\begin{aligned} &\min_{\alpha \geq 0} \ \hat{m}_k(-\alpha g_k) = f(x_k) - \alpha \|g_k\|^2 + \frac{\alpha^2}{2} \langle g_k, H_k[g_k] \rangle \\ &\text{such that } \alpha \|g_k\| \leq \Delta_k. \end{aligned}$$

This corresponds to minimizing a quadratic in α over the interval $[0, \Delta_k/\|g_k\|]$. The optimal value is easily seen to be (Conn *et al.*, 2000)

$$\alpha_k^{\mathrm{C}} = \begin{cases} \min\left(\frac{\|g_k\|^2}{\langle g_k, H_k[g_k] \rangle}, \frac{\Delta_k}{\|g_k\|}\right) & \text{if } \langle g_k, H_k[g_k] \rangle > 0, \\ \frac{\Delta_k}{\|g_k\|} & \text{otherwise.} \end{cases}$$

Lemma 3.7 Let $g_k = \operatorname{grad} f(x_k)$. Under Assumption 3.4, setting η_k to be the Cauchy step $\eta_k^{\mathrm{C}} = -\alpha_k^{\mathrm{C}} g_k$ for first-order steps fulfills Assumption 3.6 with $c_2 = 1/2$. Computing η_k^{C} involves one gradient evaluation and one application of H_k.

Proof. The claim follows as an exercise from $\hat{m}_k\left(0_{x_k}\right) - \hat{m}_k\left(\eta_k^{\mathrm{C}}\right) = \alpha_k^{\mathrm{C}} \|g_k\|^2 - \frac{(\alpha_k^{\mathrm{C}})^2}{2} \langle g_k, H_k[g_k] \rangle$ and $\langle g_k, H_k[g_k] \rangle \leq c_0 \|g_k\|^2$ owing to Assumption 3.4. □

The Steihaug–Toint truncated conjugate gradient method (Toint, 1981; Steihaug, 1983) is a monotonically improving iterative method for the trust-region subproblem whose first iterate is the Cauchy step; as such, it necessarily achieves the required model decrease.

For second-order steps the requirement is as follows.

ASSUMPTION 3.8 There exists $c_3 > 0$ such that, for all k such that $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$ and $\lambda_{\min}(H_k) < -\varepsilon_H$, the step η_k satisfies

$$\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k) \geq c_3 \Delta_k^2 \varepsilon_H. \tag{3.9}$$

This can be achieved by making a step of maximal length along a direction that certifies that $\lambda_{\min}(H_k) < -\varepsilon_H$ (Conn *et al.*, 2000): this is called an *eigenstep*. Like Cauchy steps, eigensteps can be computed in a finite number of operations, independently of ε_g and ε_H.

LEMMA 3.3 Under Assumption 3.5, if $\lambda_{\min}(H_k) < -\varepsilon_H$, there exists a tangent vector $u_k \in \mathrm{T}_{x_k}\mathcal{M}$ with

$$\|u_k\| = 1, \qquad \langle u_k, \operatorname{grad} f(x_k)\rangle \leq 0 \qquad \text{and} \qquad \langle u_k, H_k[u_k]\rangle < -\varepsilon_H.$$

Setting η_k to be any eigenstep $\eta_k^E = \Delta_k u_k$ for second-order steps fulfills Assumption 3.8 with $c_3 = 1/2$.

Let $v_1, \ldots, v_n$ be an orthonormal basis of $\mathrm{T}_{x_k}\mathcal{M}$, where $n = \dim \mathcal{M}$. One way of computing η_k^E involves the application of H_k to $v_1, \ldots, v_n$ plus $\mathcal{O}(n^3)$ arithmetic operations. The amount of work is independent of ε_g and ε_H.

Proof. Compute H, a symmetric matrix of size n that represents H_k in the basis $v_1, \ldots, v_n$, as $H_{ij} = \langle v_i, H_k[v_j]\rangle$. Compute a factorization $LDL^{\mathrm{T}} = H + \varepsilon_H I$ where I is the identity matrix, L is invertible and triangular and D is block diagonal with blocks of size 1×1 and 2×2. The factorization can be computed in $\mathcal{O}(n^3)$ operations (Golub & Van Loan, 2012, Sect. 4.4)—see the reference for a word of caution regarding pivoting for stability; pivoting is easily incorporated in the present argument. The matrix D has the same inertia as $H + \varepsilon_H I$, hence D is not positive semidefinite (otherwise $H \succcurlyeq -\varepsilon_H I$.) The structure of D makes it easy to find $x \in \mathbb{R}^n$ with $x^{\mathrm{T}} Dx < 0$. Solve the triangular system $L^{\mathrm{T}} y = x$ for $y \in \mathbb{R}^n$. Now $0 > x^{\mathrm{T}} Dx = y^{\mathrm{T}} LDL^{\mathrm{T}} y = y^{\mathrm{T}} (H + \varepsilon_H I)y$. Consequently, $y^{\mathrm{T}} Hy < -\varepsilon_H \|y\|^2$. We can set $u_k = \pm \sum_{i=1}^n y_i v_i / \|y\|$, where the sign is chosen to ensure $\langle u_k, \operatorname{grad} f(x_k)\rangle \leq 0$. To conclude check that $\hat{m}_k\left(0_{x_k}\right) - \hat{m}_k\left(\eta_k^E\right) = -\left\langle \eta_k^E, \operatorname{grad} f(x_k)\right\rangle - \frac{1}{2}\left\langle \eta_k^E, H_k\left[\eta_k^E\right]\right\rangle \geq \frac{1}{2}\Delta_k^2 \varepsilon_H$. □

Notice from the proof that this strategy either certifies that $\lambda_{\min}(H_k) \geq -\varepsilon_H \operatorname{Id}$ (which must be checked at step 8 in Algorithm 3) or certifies otherwise by providing an escape direction. We further note that, in practice, one usually prefers to use iterative methods to compute an approximate leftmost eigenvector of H_k without representing it as a matrix.

3.4 *Main results and proofs for RTR*

Under the discussed assumptions, we now establish our main theorem about computation of approximate first- and second-order critical points for (P) using RTR in a bounded number of iterations. The following constants will be useful:

$$\lambda_g = \frac{1}{4}\min\left(\frac{1}{c_0}, \frac{c_2}{L_g + c_0}\right) \qquad \text{and} \qquad \lambda_H = \frac{3}{4}\frac{c_3}{L_H + c_1}. \tag{3.10}$$

THEOREM 3.9 Under Assumptions 2.3, 3.1, 3.4, 3.6 and assuming $\varepsilon_g \leq \frac{\Delta_0}{\lambda_g}$,[7] Algorithm 3 produces an iterate x_{N_1} satisfying $\|\operatorname{grad} f(x_{N_1})\| \leq \varepsilon_g$ with

$$N_1 \leq \frac{3}{2}\frac{f(x_0) - f^*}{\rho' c_2 \lambda_g}\frac{1}{\varepsilon_g^2} + \frac{1}{2}\log_2\left(\frac{\Delta_0}{\lambda_g \varepsilon_g}\right) = \mathcal{O}\left(\frac{1}{\varepsilon_g^2}\right). \tag{3.11}$$

Furthermore, if $\varepsilon_H < \infty$ then under additional Assumptions 3.2, 3.5, and 3.8 and assuming $\varepsilon_g \leq \frac{c_2}{c_3}\frac{\lambda_H}{\lambda_g^2}$ and $\varepsilon_H \leq \frac{c_2}{c_3}\frac{1}{\lambda_g}$, Algorithm 3 also produces an iterate x_{N_2} satisfying $\|\operatorname{grad} f(x_{N_2})\| \leq \varepsilon_g$ and $\lambda_{\min}(H_{N_2}) \geq -\varepsilon_H$ with

$$N_1 \leq N_2 \leq \frac{3}{2}\frac{f(x_0) - f^*}{\rho' c_3 \lambda^2}\frac{1}{\varepsilon^2 \varepsilon_H} + \frac{1}{2}\log_2\left(\frac{\Delta_0}{\lambda\varepsilon}\right) = \mathcal{O}\left(\frac{1}{\varepsilon^2 \varepsilon_H}\right), \tag{3.12}$$

where we defined $(\lambda, \varepsilon) = (\lambda_g, \varepsilon_g)$ if $\lambda_g \varepsilon_g \leq \lambda_H \varepsilon_H$ and $(\lambda, \varepsilon) = (\lambda_H, \varepsilon_H)$ otherwise. Since the algorithm is a descent method, $f(x_{N_2}) \leq f(x_{N_1}) \leq f(x_0)$.

REMARK 3.10 Theorem 3.4 makes a statement about $\lambda_{\min}(H_k)$ at termination, *not* about $\lambda_{\min}(\operatorname{Hess} f(x_k))$. See Section 3.5 to connect these two quantities.

To establish Theorem 3.4 we work through a few lemmas, following the proof technique in Cartis *et al.* (2012). We first show Δ_k is bounded below in proportion to the tolerances ε_g and ε_H. This is used to show that the number of successful iterations in Algorithm 3 before termination (that is, iterations where $\rho_k > \rho'$; see (3.3)) is bounded above. It is then shown that the total number of iterations is at most a constant multiple of the number of successful iterations, which implies termination in bounded time.

We start by showing that the trust-region radius is bounded away from zero. Essentially, this is because if Δ_k becomes too small, then the Cauchy step and eigenstep are certain to be successful owing to the quality of the model in such a small region, so that the trust-region radius could not decrease any further.

LEMMA 3.11 Under the assumptions of Theorem 3.4, if Algorithm 3 executes N iterations without terminating then

$$\Delta_k \geq \min\left(\Delta_0, \lambda_g \varepsilon_g, \lambda_H \varepsilon_H\right) \tag{3.13}$$

for $k = 0, \ldots, N$, where λ_g and λ_H are defined in (3.10).

Proof. This follows essentially the proof of Absil *et al.* (2008, Thm. 7.4.2), which itself follows classical proofs (Conn *et al.*, 2000). The core idea is to control ρ_k (see (3.3)) close to 1, to show that there cannot be arbitrarily many trust-region radius reductions. The proof is in two parts.

[7] Theorem 3.4 is scale invariant, in that if the cost function $f(x)$ is replaced by $\alpha f(x)$ for some positive α (which does not meaningfully change (P)), it is sensible to also multiply L_g, L_H, c_0, c_1, ε_g and ε_H by α; consequently, the upper bounds on ε_g and ε_H and the upper bounds on N_1 and N_2 are invariant under this scaling. If it is desirable to always allow ε_g, ε_H in, say, $(0, 1]$, one possibility is to artificially make L_g, L_H, c_0, c_1 larger (which is always allowed).

For the first part, assume $\|\operatorname{grad} f(x_k)\| > \varepsilon_g$. Then consider the gap

$$|\rho_k - 1| = \left| \frac{\hat{f}_k(0_{x_k}) - \hat{f}_k(\eta_k)}{\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k)} - 1 \right| = \left| \frac{\hat{m}_k(\eta_k) - \hat{f}_k(\eta_k)}{\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k)} \right|. \tag{3.14}$$

From Assumption 3.6, we know the denominator is not too small:

$$\hat{m}_k\left(0_{x_k}\right) - \hat{m}_k(\eta_k) \geq c_2 \min\left(\Delta_k, \frac{\varepsilon_g}{c_0}\right) \varepsilon_g.$$

Now consider the numerator:

$$\begin{aligned} |\hat{m}_k(\eta_k) - \hat{f}_k(\eta_k)| &= \left| f(x_k) + \langle \operatorname{grad} f(x_k), \eta_k \rangle + \tfrac{1}{2} \langle \eta_k, H_k[\eta_k] \rangle - \hat{f}_k(\eta_k) \right| \\ &\leq \left| f(x_k) + \langle \operatorname{grad} f(x_k), \eta_k \rangle - \hat{f}_k(\eta_k) \right| + \tfrac{1}{2} \left| \langle \eta_k, H_k[\eta_k] \rangle \right| \\ &\leq \tfrac{1}{2} \left(L_g + c_0 \right) \|\eta_k\|^2, \end{aligned}$$

where we used Assumption 3.1 for the first term, and Assumption 3.4 for the second term. Assume for the time being that $\Delta_k \leq \min\left(\frac{\varepsilon_g}{c_0}, \frac{c_2 \varepsilon_g}{L_g + c_0}\right) = 4\lambda_g \varepsilon_g$. Then, using $\|\eta_k\| \leq \Delta_k$, it follows that

$$|\rho_k - 1| \leq \frac{1}{2} \frac{L_g + c_0}{c_2 \min\left(\Delta_k, \frac{\varepsilon_g}{c_0}\right) \varepsilon_g} \Delta_k^2 \leq \frac{1}{2} \frac{L_g + c_0}{c_2 \varepsilon_g} \Delta_k \leq \frac{1}{2}.$$

Hence, $\rho_k \geq 1/2$, and by the mechanism of Algorithm 3, it follows that $\Delta_{k+1} \geq \Delta_k$.

For the second part, assume $\|\operatorname{grad} f(x_k)\| < \varepsilon_g$ and $\lambda_{\min}(H_k) < -\varepsilon_H$. Then, by Assumption 3.8,

$$\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k) \geq c_3 \Delta_k^2 \varepsilon_H.$$

Thus, by Assumptions 3.2 and 3.5,

$$\begin{aligned} |\hat{m}_k(\eta_k) - \hat{f}_k(\eta_k)| &= \left| f(x_k) + \langle \operatorname{grad} f(x_k), \eta_k \rangle + \tfrac{1}{2} \langle \eta_k, H_k[\eta_k] \rangle - \hat{f}_k(\eta_k) \right| \\ &\leq \frac{L_H}{6} \|\eta_k\|^3 + \tfrac{1}{2} \left| \left\langle \eta_k, \left(\nabla^2 \hat{f}_k\left(0_{x_k}\right) - H_k \right) [\eta_k] \right\rangle \right| \\ &\leq \frac{L_H + c_1}{6} \Delta_k^3. \end{aligned}$$

As previously, combine these observations into (3.14) to see that if $\Delta_k \leq \frac{3c_3}{L_H + c_1} \varepsilon_H = 4\lambda_H \varepsilon_H$ then

$$|\rho_k - 1| \leq \tfrac{1}{2} \frac{L_H + c_1}{3c_3 \varepsilon_H} \Delta_k \leq \tfrac{1}{2}. \tag{3.15}$$

Again, this implies $\Delta_{k+1} \geq \Delta_k$.

Now combine the two parts. We have established that, if $\Delta_k \leq 4\min\left(\lambda_g\varepsilon_g, \lambda_H\varepsilon_H\right)$, then $\Delta_{k+1} \geq \Delta_k$. To conclude the proof, consider the fact that Algorithm 3 cannot reduce the radius by more than 1/4 in one step. □

By an argument similar to the one used for gradient methods, Lemma 3.6 implies an upper bound on the number of successful iterations required in Algorithm 3 to reach termination.

LEMMA 3.12 Under the assumptions of Theorem 3.4, if Algorithm 3 executes N iterations without terminating, define the set of *successful steps* as

$$S_N = \{k \in \{0, \ldots, N\} : \rho_k > \rho'\}$$

and let U_N designate the unsuccessful steps, so that S_N and U_N form a partition of $\{0, \ldots, N\}$. Assume $\varepsilon_g \leq \Delta_0/\lambda_g$. If $\varepsilon_H = \infty$, the number of successful steps obeys

$$|S_N| \leq \frac{f(x_0) - f^*}{\rho' c_2 \lambda_g} \frac{1}{\varepsilon_g^2}. \tag{3.16}$$

Otherwise, if additionally $\varepsilon_g \leq \frac{c_2}{c_3}\frac{\lambda_H}{\lambda_g^2}$ and $\varepsilon_H \leq \frac{c_2}{c_3}\frac{1}{\lambda_g}$, we have the bound

$$|S_N| \leq \frac{f(x_0) - f^*}{\rho' c_3} \frac{1}{\min(\lambda_g\varepsilon_g, \lambda_H\varepsilon_H)^2\varepsilon_H}. \tag{3.17}$$

Proof. The proof parallels Cartis *et al.* (2012, Lemma 4.5). Clearly, if $k \in U_N$, then $f(x_k) = f(x_{k+1})$. On the other hand, if $k \in S_N$ then $\rho_k \geq \rho'$ (see (3.3)). Combine this with Assumptions 3.6 and 3.8 to see that, for $k \in S_N$,

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \rho'\left(\hat{m}_k(0_{x_k}) - \hat{m}_k(\eta_k)\right) \\ &\geq \rho' \min\left(c_2 \min\left(\Delta_k, \frac{\varepsilon_g}{c_0}\right)\varepsilon_g \;\; c_3\Delta_k^2\varepsilon_H\right). \end{aligned}$$

By Lemma 3.6 and the assumption $\lambda_g\varepsilon_g \leq \Delta_0$, it holds that $\Delta_k \geq \min\left(\lambda_g\varepsilon_g, \lambda_H\varepsilon_H\right)$. Furthermore, using $\lambda_g \leq 1/c_0$ shows that $\min(\Delta_k, \varepsilon_g/c_0) \geq \min(\Delta_k, \lambda_g\varepsilon_g) \geq \min\left(\lambda_g\varepsilon_g, \lambda_H\varepsilon_H\right)$. Hence,

$$f(x_k) - f(x_{k+1}) \geq \rho' \min\left(c_2\lambda_g\varepsilon_g^2, c_2\lambda_H\varepsilon_g\varepsilon_H, c_3\lambda_g^2\varepsilon_g^2\varepsilon_H, c_3\lambda_H^2\varepsilon_H^3\right). \tag{3.18}$$

If $\varepsilon_H = \infty$ this simplifies to

$$f(x_k) - f(x_{k+1}) \geq \rho' c_2 \lambda_g \varepsilon_g^2.$$

Sum over iterations up to N and use Assumption 2.3 (bounded f):

$$f(x_0) - f^* \geq f(x_0) - f(x_{N+1}) = \sum_{k \in S_N} f(x_k) - f(x_{k+1}) \geq |S_N| \rho' c_2 \lambda_g \varepsilon_g^2.$$

Hence,

$$|S_N| \leq \frac{f(x_0) - f^*}{\rho' c_2 \lambda_g} \frac{1}{\varepsilon_g^2}.$$

On the other hand, if $\varepsilon_H < \infty$ then, starting over from (3.18) and assuming both $c_3 \lambda_g^2 \varepsilon_g^2 \varepsilon_H \leq c_2 \lambda_H \varepsilon_g \varepsilon_H$ and $c_3 \lambda_g^2 \varepsilon_g^2 \varepsilon_H \leq c_2 \lambda_g \varepsilon_g^2$ (which is equivalent to $\varepsilon_g \leq c_2 \lambda_H / c_3 \lambda_g^2$ and $\varepsilon_H \leq c_2 / c_3 \lambda_g$), it comes with the same telescoping sum that

$$f(x_0) - f^* \geq |S_N| \rho' c_3 \min(\lambda_g \varepsilon_g, \lambda_H \varepsilon_H)^2 \varepsilon_H.$$

Solve for $|S_N|$ to conclude. □

Finally, we show that the total number of steps N before termination cannot be more than a fixed multiple of the number of successful steps $|S_N|$.

LEMMA 3.13 Under the assumptions of Theorem 3.4, if Algorithm 3 executes N iterations without terminating, using the notation S_N and U_N of Lemma 3.7, it holds that

$$|S_N| \geq \tfrac{2}{3}(N+1) - \tfrac{1}{3} \max\left(0, \log_2\left(\frac{\Delta_0}{\lambda_g \varepsilon_g}\right), \log_2\left(\frac{\Delta_0}{\lambda_H \varepsilon_H}\right)\right). \tag{3.19}$$

Proof. The proof rests on the lower bound for Δ_k obtained in Lemma 3.6. It parallels Cartis *et al.* (2012, Lemma 4.6). For all $k \in S_N$, it holds that $\Delta_{k+1} \leq 2\Delta_k$. For all $k \in U_k$, it holds that $\Delta_{k+1} \leq \frac{1}{4}\Delta_k$. Hence,

$$\Delta_N \leq 2^{|S_N|} \left(\tfrac{1}{4}\right)^{|U_N|} \Delta_0.$$

On the other hand, Lemma 3.6 gives

$$\Delta_N \geq \min\left(\Delta_0, \lambda_g \varepsilon_g, \lambda_H \varepsilon_H\right).$$

Combine, divide by Δ_0 and take the log in base 2:

$$|S_N| - 2|U_N| \geq \min\left(0, \log_2\left(\frac{\lambda_g \varepsilon_g}{\Delta_0}\right), \log_2\left(\frac{\lambda_H \varepsilon_H}{\Delta_0}\right)\right).$$

Use $|S_N| + |U_N| = N + 1$ to conclude. □

We can now prove the main theorem.

Proof of Theorem 3.4. It is sufficient to combine Lemmas 3.6 and 3.7 in both regimes. First, we get that if $\|\operatorname{grad} f(x_k)\| > \varepsilon_g$ for $k = 0, \ldots, N$, then

$$N + 1 \leq \tfrac{3}{2} \frac{f(x_0) - f^*}{\rho' c_2 \lambda_g} \frac{1}{\varepsilon_g^2} + \tfrac{1}{2} \log_2\left(\frac{\Delta_0}{\lambda_g \varepsilon_g}\right).$$

(The term $\log_2\left(\frac{\Delta_0}{\lambda_H \varepsilon_H}\right)$ from Lemma 3.8 is irrelevant up to that point, as ε_H could just as well have been infinite.) Thus, after a number of iterations larger than the right-hand side, an iterate with sufficiently small gradient must have been produced, to avoid a contradiction.

Second, we get that if for $k = 0, \ldots, N$ no iterate satisfies both $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$ and $\lambda_{\min}(H_k) \geq -\varepsilon_H$, then

$$N + 1 \leq \tfrac{3}{2}\frac{f(x_0) - f^*}{\rho' c_3}\frac{1}{\min(\lambda_g \varepsilon_g, \lambda_H \varepsilon_H)^2 \varepsilon_H} + \tfrac{1}{2}\max\left(\log_2\left(\frac{\Delta_0}{\lambda_g \varepsilon_g}\right), \log_2\left(\frac{\Delta_0}{\lambda_H \varepsilon_H}\right)\right).$$

Conclude with the same argument. □

3.5 *Connecting H_k and Hess $f(x_k)$*

Theorem 3.4 states termination of Algorithm 3 in terms of $\|\operatorname{grad} f(x_k)\|$ and $\lambda_{\min}(H_k)$. Ideally, the latter must be turned into a statement about $\lambda_{\min}(\operatorname{Hess} f(x_k))$, to match the second-order necessary optimality conditions of (P) more closely (recall Proposition 1.1). Assumption 3.5 itself requires only H_k to be (weakly) related to $\nabla^2 \hat{f}_k(0_{x_k})$ (the Hessian of the pullback of f at x_k), which is different from the Riemannian Hessian of f at x_k in general. It is up to the user to provide H_k sufficiently related to $\nabla^2 \hat{f}_k\left(0_{x_k}\right)$. Additional control over the retraction at x_k can further relate $\nabla^2 \hat{f}_k\left(0_{x_k}\right)$ to $\operatorname{Hess} f(x_k)$, as we do now. Proofs for this section are in Appendix D.

LEMMA 3.9 Define the maximal acceleration of Retr at x as the real a such that

$$\forall\, \eta \in \mathrm{T}_x\mathcal{M} \text{ with } \|\eta\| = 1, \quad \left\| \frac{\mathrm{D}^2}{\mathrm{d}t^2} \operatorname{Retr}_x(t\eta)|_{t=0} \right\| \leq a,$$

where $\frac{\mathrm{D}^2}{\mathrm{d}t^2}\gamma$ denotes acceleration of the curve $t \mapsto \gamma(t)$ on $\mathcal{M}$ (Absil *et al.*, 2008, Chapter 5). Then

$$\left\| \operatorname{Hess} f(x) - \nabla^2 \hat{f}_x(0_x) \right\| \leq a \cdot \|\operatorname{grad} f(x)\|.$$

In particular, if x is a critical point or if $a = 0$, the Hessians agree: $\operatorname{Hess} f(x) = \nabla^2 \hat{f}_x(0_x)$.

The particular cases appear as Absil *et al.* (2008, Props. 5.5.5, 5.5.6). This result highlights the crucial role of retractions with zero acceleration, known as *second-order retractions* and defined in Absil *et al.* (2008, Prop. 5.5.5); we are not aware of earlier references to this notion.

DEFINITION 3.14 A retraction is a *second-order retraction* if it has zero acceleration, as defined in Lemma 3.9. Then retracted curves locally agree with geodesics up to second order.

PROPOSITION 3.15 Let $x_k \in \mathcal{M}$ be the iterate returned by Algorithm 3 under the assumptions of Theorem 3.4. It satisfies $\|\operatorname{grad} f(x_k)\| \leq \varepsilon_g$ and $H_k \succcurlyeq -\varepsilon_H \operatorname{Id}$. Assume H_k is related to the Hessian of the pullback as $\|\nabla^2 \hat{f}_k\left(0_{x_k}\right) - H_k\| \leq \delta_k$. Further assume the retraction has acceleration at x_k bounded by a_k, as defined in Lemma 3.9. Then

$$\operatorname{Hess} f(x_k) \succeq -\left(\varepsilon_H + a_k \varepsilon_g + \delta_k\right) \operatorname{Id}.$$

In particular, if the retraction is second order and $H_k = \nabla^2 \hat{f}_k\left(0_{x_k}\right)$, then $\operatorname{Hess} f(x_k) \succcurlyeq -\varepsilon_H \operatorname{Id}$.

We note that second-order retractions are frequently available in applications. Indeed, retractions for submanifolds obtained as (certain types of) projections—arguably one of the most natural classes of retractions for submanifolds—are second order (Absil & Malick, 2012, Thm. 22). For example, the sphere retraction $\mathrm{Retr}_x(\eta) = (x + \eta)/\|x + \eta\|$ is second order. Such retractions for low-rank matrices are also known (Absil & Oseledets, 2015).

4. Example: smooth semidefinite programs

This example is based on Boumal *et al.* (2016). Consider the following semidefinite program, which occurs in robust PCA (McCoy & Tropp, 2011) and as a convex relaxation of combinatorial problems such as Max-Cut, $\mathbb{Z}_2$-synchronization and community detection in the stochastic block model (Goemans & Williamson, 1995; Bandeira *et al.*, 2016):

$$\min_{X \in \mathbb{R}^{n \times n}} \mathrm{Tr}(CX) \text{ subject to } \mathrm{diag}(X) = \mathbf{1}, X \succeq 0. \tag{4.1}$$

The symmetric cost matrix C depends on the application. Interior point methods solve this problem in polynomial time, though they involve significant work to enforce the conic constraint $X \succcurlyeq 0$ (X symmetric, positive semidefinite). This motivates the approach of Burer & Monteiro (2005) to parameterize the search space as $X = YY^{\mathrm{T}}$, where Y is in $\mathbb{R}^{n \times p}$ for some well-chosen p:

$$\min_{Y \in \mathbb{R}^{n \times p}} \mathrm{Tr}\left(CYY^{\mathrm{T}}\right) \text{ subject to } \mathrm{diag}\left(YY^{\mathrm{T}}\right) = \mathbf{1}. \tag{4.2}$$

This problem is of the form of (P), where $f(Y) = \mathrm{Tr}\left(CYY^{\mathrm{T}}\right)$ and the manifold is a product of n unit spheres in $\mathbb{R}^p$:

$$\mathcal{M} = \left\{Y \in \mathbb{R}^{n \times p} : \mathrm{diag}\left(YY^{\mathrm{T}}\right) = \mathbf{1}\right\} = \left\{Y \in \mathbb{R}^{n \times p} : \text{each row of Y has unit norm}\right\}. \tag{4.3}$$

In principle, since the parameterization $X = YY^{\mathrm{T}}$ breaks convexity, the new problem could have many spurious local optimizers and saddle points. Yet, for $p = n + 1$, it has recently been shown that approximate second-order critical points Y map to approximate global optimizers $X = YY^{\mathrm{T}}$, as stated in the following proposition. (In this particular case there is no need to control $\|\mathrm{grad}\, f(Y)\|$ explicitly.)

Proposition 4.1 (Boumal *et al.*, 2016). If $X^\star$ is optimal for (3.19) and Y is feasible for (4.1) with $p > n$ and $\mathrm{Hess}\, f(Y) \succcurlyeq -\varepsilon_H \,\mathrm{Id}$, the optimality gap is bounded as

$$0 \leq \mathrm{Tr}\left(CYY^{\mathrm{T}}\right) - \mathrm{Tr}(CX^\star) \leq \frac{n}{2}\varepsilon_H.$$

Since f is smooth in $\mathbb{R}^{n \times p}$ and $\mathcal{M}$ is a compact submanifold of $\mathbb{R}^{n \times p}$, the regularity Assumptions 3.1 and 3.2 hold with any second-order retraction (Lemmas 2.4 and 3.1). In particular, they hold if $\mathrm{Retr}_Y(\dot{Y})$ is the result of normalizing each row of $Y + \dot{Y}$ (Section 3.5) or if the exponential map is used (which is cheap for this manifold; see Appendix E). Theorem 3.4 then implies that RTR applied to the nonconvex problem (4.2) computes a point $X = YY^{\mathrm{T}}$ feasible for (4.1) such that $\mathrm{Tr}(CX) - \mathrm{Tr}(CX^\star) \leq \delta$ in $\mathcal{O}\left(1/\delta^3\right)$ iterations. Appendix E bounds the total work with an explicit dependence on the problem dimension n as $\mathcal{O}\left(n^{10}/\delta^3\right)$ arithmetic operations, where $\mathcal{O}$ hides factors depending on the data C and an additive log

term. This result follows from a bound $L_H \leq 8\,\|C\|_2\,\sqrt{n}$ for Assumption 3.2, which is responsible for a factor of n in the complexity—the remaining factors could be improved; see below.

In Boumal *et al.* (2016), it is shown that, generically in C, if $p \geq \lceil \sqrt{2n} \rceil$ then all second-order critical points of (4.2) are globally optimal (despite nonconvexity). This means RTR globally converges to global optimizers with cheaper iterations (due to reduced dimensionality). Unfortunately, there is no statement of quality pertaining to *approximate* second-order critical points for such small p, so that this analysis is not sufficient to obtain an improved worst-case complexity bound.

These bounds are worse than guarantees provided by interior point methods. Indeed, following Nesterov (2004, Sect. 4.3.3, with eq. (4.3.12)), interior point methods achieve a solution in $\mathcal{O}\big(n^{3.5}\log(n/\delta)\big)$ arithmetic operations. Yet, numerical experiments in Boumal *et al.* (2016) suggest RTR often outperforms interior point methods, indicating that the bound $\mathcal{O}\big(n^{10}/\delta^3\big)$ is wildly pessimistic. We report it here mainly because, to the best of our knowledge, this is the first explicit bound for a Burer–Monteiro approach to solving a semidefinite program.

A number of factors drive the gap between our worst-case bound and practice. In particular, strategies far more efficient than the LDL^{T} factorization in Lemma 3.3 are used to compute second-order steps, and they can exploit structure in C. High-accuracy solutions are reached owing to RTR typically converging superlinearly, locally. And p is chosen much smaller than $n + 1$.

See also Mei *et al.* (2017) for formal complexity results in a setting where p is allowed to be independent of n; this precludes reaching an objective value arbitrarily close to optimal, in exchange for cheaper computations.

5. Conclusions and perspectives

We presented bounds on the number of iterations required by the Riemannian gradient descent algorithm and the RTR algorithm to reach points that approximately satisfy first- and second-order necessary optimality conditions, under some regularity assumptions but regardless of initialization. When the search space $\mathcal{M}$ is a Euclidean space these bounds are already known. For the more general case of $\mathcal{M}$ being a Riemannian manifold, these bounds are new.

As a subclass of interest, we showed the regularity requirements are satisfied if $\mathcal{M}$ is a compact submanifold of $\mathbb{R}^n$ and f has locally Lipschitz continuous derivatives of appropriate order. This covers a rich class of practical optimization problems.

While there are no explicit assumptions made about $\mathcal{M}$, the smoothness requirements for the pullback of the cost—Assumptions 2.6, 3.1 and 3.2—implicitly restrict the class of manifolds to which these results apply. Indeed, for certain manifolds, even for nice cost functions f, there may not exist retractions that ensure the assumptions hold. This is the case in particular for certain incomplete manifolds, such as open Riemannian submanifolds of $\mathbb{R}^n$ and certain geometries of the set of fixed-rank matrices—see also Remark 2.2 about injectivity radius. For such sets it may be necessary to adapt the assumptions. For fixed-rank matrices for example, Vandereycken (2013, Sect. 4.1) obtains convergence results assuming a kind of coercivity on the cost function: for any sequence of rank-k matrices $(X_i)_{i=1,2,\ldots}$ such that the first singular value $\sigma_1(X_i) \to \infty$ or the kth singular value $\sigma_k(X_i) \to 0$, it holds that $f(X_i) \to \infty$. This ensures iterates stay away from the open boundary.

The iteration bounds are sharp, but additional information may yield more favorable bounds in specific contexts. In particular, when the studied algorithms converge to a nondegenerate local optimizer, they do so with an at least linear rate, so that the number of iterations is merely $\mathcal{O}(\log(1/\varepsilon))$ once in the linear regime. This suggests a stitching approach: for a given application, it may be possible to show that rough approximate second-order critical points are in a local attraction basin; the iteration cost can

then be bounded by the total work needed to attain such a crude point starting from anywhere, plus the total work needed to refine the crude point to high accuracy with a linear or even quadratic convergence rate. This is, to some degree, the successful strategy in Sun *et al.* (2015, 2016).

Finally, we note that it would also be interesting to study the global convergence rates of Riemannian versions of adaptive regularization algorithms using cubics (ARC), since in the Euclidean case these can achieve approximate first-order criticality in $\mathcal{O}\left(1/\varepsilon^{1.5}\right)$ instead of $\mathcal{O}\left(1/\varepsilon^{2}\right)$ (Cartis *et al.*, 2011a). Work in that direction could start with the convergence analyses proposed in Qi, (2011).

Acknowledgements

This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme initiated by the Belgian Science Policy Office. This work was supported by the ARC 'Mining and Optimization of Big Data Models'. We thank Alex d'Aspremont, Simon Lacoste-Julien, Ju Sun, Bart Vandereycken and Paul Van Dooren for helpful discussions.

Funding

Fonds Spéciaux de Recherche (FSR) at UCLouvain (to N.B.); Chaire Havas 'Chaire Economie et gestion des nouvelles données' (to N.B.); ERC Starting Grant SIPA (to N.B.); Research in Paris grant at Inria & ENS (to N.B.); National Science Foundation (DMS-1719558 to N.B.); The Natural Environment Research Council (grant NE/L012146/1 to C.C.).

References

Absil, P.-A., Baker, C. G. & Gallivan, K. A. (2007) Trust-region methods on Riemannian manifolds. *Found. Comput. Math*, **7**, 303–330. doi:10.1007/s10208–005-0179–9.

Absil, P.-A., Mahony, R. & Sepulchre, R. (2008) *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ: Princeton University Press. ISBN978–0-691–13298-3.

Absil, P.-A., Mahony, R. & Trumpf, J. (2013) An extrinsic look at the Riemannian Hessian. *Geometric Science of Information* (F. Nielsen and F. Barbaresco eds). Lecture Notes in Computer Science, vol. **8085**. Berlin Heidelberg: Springer, pp 361–368. ISBN 978–3-642–40019-3. doi:10.1007/978–3-642–40020-9 39. URL http://sites.uclouvain.be/absil/2013.01

Absil, P.-A. & Malick, J. (2012) Projection-like retractions on matrix manifolds. *SIAM J. Optim.*, **22**, 135–158. doi:10.1137/100802529.

Absil, P.-A. & Oseledets, I. (2015) Low-rank retractions: a survey and new results. *Comput. Optim. Appl.*, **62**, 5–29. doi:10.1007/s10589–014-9714–4. Accepted for publication.

Absil, P.-A., Trumpf, J., Mahony, R. & Andrews, B. (2009) All roads lead to Newton: feasible second-order methods for equality-constrained optimization. *Technical Report UCL-INMA-2009.024*. Belgium: Departement d'ingenierie mathematique, UCLouvain.

Adler, R., Dedieu, J., Margulies, J., Martens, M. & Shub, M. (2002) Newton's method on Riemannian manifolds and a geometric model for the human spine. *IMA J. Numer. Anal.*, **22**, 359–390. doi:10.1093/imanum/22.3.359.

Bandeira, A., Boumal, N. & Voroninski, V. (2016) On the low-rank approach for semidefinite programs arising in synchronization and community detection. *Proceedings of the 29th Conference on Learning Theory, COLT 2016*. New York: PMLR, June 23–26.

Bento, G., Ferreira, O. & Melo, J. (2017) Iteration-complexity of gradient, subgradient and proximal point methods on Riemannian manifolds. *J. Optim. Theory Appl.*, **173**, 548–562. doi:10.1007/s10957–017-1093–4.

Berger, G. (2017) Fast matrix multiplication. Master Thesis. Ecole polytechnique de Louvain. Available at http://hdl.handle.net/2078.1/thesis:10630

Bhojanapalli, S., Neyshabur, B. & Srebro, N. (2016) Global optimality of local search for low rank matrix recovery. Preprint arXiv:1605. 07221.

Birgin, E., Gardenghi, J., Martínez, J., Santos, S. & Toint, P. (2017) Worst-case evaluation complexity for unconstrained nonlinear optimization using high-order regularized models. *Math. Prog.*, **163**, 359–368. doi:10.1007/s10107–016-1065–8.

Boumal, N. (2015b) A Riemannian low-rank method for optimization over semidefinite matrices with block-diagonal constraints. Preprint arXiv:1506.00575.

Boumal, N. (2015a) Riemannian trust regions with finite-difference Hessian approximations are globally convergent. *Geometric Science of Information* (F. Nielsen and F. Barbaresco eds). Lecture Notes in Computer Science, vol. **9389**. Springer International Publishing, pp. 467–475. doi:10.1007/978–3-319–25040-3 50.

Boumal, N. (2016) Nonconvex phase synchronization. *SIAM J. Optim.*, **26**, 2355–2377. doi:10.1137/16M105808X.

Boumal, N., Mishra, B., Absil, P.-A. & Sepulchre, R. (2014) Manopt, a Matlab toolbox for optimization on manifolds. *J. Mach. Learn. Res.*, **15**, 1455–1459. (http://www.manopt.org)

Boumal, N., Voroninski, V. & Bandeira, A. (2016) The non-convex Burer-Monteiro approach works on smooth semidefinite programs. *Advances in Neural Information Processing Systems* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett eds). Vol. **29**, Curran Associates, pp. 2757–2765.

Burer, S. & Monteiro, R. (2005) Local minima and convergence in low-rank semidefinite programming. *Math. Prog.*, **103**, 427–444.

Cartis, C., Gould, N. I. M. & Toint, P. L. (2010) On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems. *SIAM J. Optim.*, **20**, 2833–2852. doi:10.1137/090774100.

Cartis, C., Gould, N. & Toint, P. (2011a) Adaptive cubic regularisation methods for unconstrained optimization. Part II: worst-case function- and derivative-evaluation complexity. *Math. Prog.*, **130**, 295–319. doi:10.1007/s10107–009-0337-y.

Cartis, C., Gould, N. & Toint, P. (2011b) Optimal Newton-type methods for nonconvex smooth optimization problems. *ERGO Technical Report 11–009.* School of Mathematics, University of Edinburgh.

Cartis, C., Gould, N. & Toint, P. (2012) Complexity bounds for second-order optimality in unconstrained optimization. *J. Complexity*, **28**, 93–108. doi:10.1016/j.jco.2011.06.001.

Cartis, C., Gould, N. & Toint, P. (2014) On the complexity of finding first-order critical points in constrained nonlinear optimization. *Math. Prog.*, **144**, 93–106. doi:10.1007/s10107–012-0617–9.

Cartis, C., Gould, N. & Toint, P. (2015a) Evaluation complexity bounds for smooth constrained nonlinear optimization using scaled KKT conditions and high-order models. *NA Technical Report, Maths E-print Archive1912*. Mathematical Institute, Oxford University.

Cartis, C., Gould, N. & Toint, P. (2015b) On the evaluation complexity of constrained nonlinear least-squares and general constrained nonlinear optimization using second-order methods. *SIAM J. Numer. Anal.*, **53**, 836–851. doi:10.1137/130915546.

Cartis, C., Gould, N. I. M. & Toint, P. L. (2017) Second-order optimality and beyond: Characterization and evaluation complexity in convexly constrained nonlinear optimization. *Found. Comput. Math*. doi:10.1007/s10208-017-9363-y.

Chavel, I. (2006) *Riemannian Geometry: A Modern Introduction*, Cambridge Tracts in Mathematics, vol. **108**. Cambridge: Cambridge University Press.

Conn, A., Gould, N. & Toint, P. (2000) *Trust-Region Methods*. MPS-SIAM Series on Optimization. SIAM. ISBN 978–0-89871–460-9. doi:10.1137/1.9780898719857.

Curtis, F. E., Robinson, D. P. & Samadi, M. (2016) A trust region algorithm with a worst-case iteration complexity of $\mathcal{O}(\epsilon^{-3/2})$ for nonconvex optimization. *Math. Prog.*, **162**, pages 1–32. doi:10.1007/s10107–016-1026–2.

Edelman, A., Arias, T. & Smith, S. (1998) The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, **20**, 303–353.

Gabay, D. (1982) Minimizing a differentiable function over a differential manifold. *J. Optim. Theory Appl.*, **37**, 177–219.

Ge, R., Lee, J. & Ma, T. (2016) Matrix completion has no spurious local minimum. *Advances in Neural Information Processing Systems* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon and R. Garnett eds). Vol. **29** Curran Associates, pp. 2973–2981. (http://papers.nips.cc/paper/6048-matrix-completion-has-no-spurious-local-minimum.pdf).

Goemans, M. & Williamson, D. (1995) Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, **42**, 1115–1145. doi:10.1145/227683.227684.

Golub, G. & Van Loan, C. (2012) *Matrix Computations*, 4th edn. Johns Hopkins Studies in the Mathematical Sciences, vol. **3**. Baltimore, Maryland: Johns Hopkins University Press. doi:10.1137/0720042.

Huang, W., Absil, P.-A., Gallivan, K. & Hand, P. (2016) ROPTLIB: an object-oriented C++ library for optimization on Riemannian manifolds. *Technical Report FSU16–14.v2*. Florida State University.

Huang, W., Gallivan, K. & Absil, P.-A. (2015) A Broyden class of quasi-Newton methods for Riemannian optimization. *SIAM J. Optim.*, **25**, 1660–1685. doi:10.1137/140955483.

McCoy, M. & Tropp, J. (2011) Two proposals for robust PCA using semidefinite programming. *Electron. J. Stat.*, **5**, 1123–1160. doi:10.1214/11-EJS636.

Mei, S., Misiakiewicz, T., Montanari, A. & Oliveira, R. (2017) Solving SDPs for synchronization and MaxCut problems via the Grothendieck inequality. Preprint arXiv:1703.08729.

Monera, M. G., Montesinos-Amilibia, A. & Sanabria-Codesal, E. (2014) The Taylor expansion of the exponential map and geometric applications. *Rev. R. Acad. Cienc. Exactas, Físicas Nat. Ser. A Math. RACSAM*, **108**, 881–906. doi:10.1007/s13398–013-0149-z.

Moré, J. & Sorensen, D. (1983) Computing a trust region step. *SIAM J. Sci. Stat. Comput.*, **4**, 553–572. doi:10.1137/0904038.

Nesterov, Y. (2004) *Introductory Lectures on Convex Optimization: A Basic Course*. Applied Optimization, vol. **87**. Boston, MA: Springer. ISBN 978–1-4020–7553-7.

Nocedal, J. & Wright, S. (1999) *Numerical Optimization*. New York: Springer.

O'Neill, B. (1983) *Semi-Riemannian Geometry: With Applications to Relativity*, vol. 103, Academic Press.

Qi, C. (2011) Numerical optimization methods on Riemannian manifolds. *PhD thesis*, Florida State University, Tallahassee, FL.

Ring, W. & Wirth, B. (2012) Optimization methods on Riemannian manifolds and their application to shape space. *SIAM J. Optim.*, **22**, 596–627. doi:10.1137/11082885X.

Ruszczyński, A. (2006) *Nonlinear Optimization*. Princeton, NJ: Princeton University Press.

Sato, H. (2016) A Dai–Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions. *Comput. Optim. Appl.,***64**, 101–118. doi:10.1007/s10589–015-9801–1.

Shub, M. (1986) Some remarks on dynamical systems and numerical analysis. *Proceedings of VII ELAM.*, (L. Lara-Carrero & J. Lewowicz, eds), Equinoccio, Univ. Simón Bolívar, Caracas, pp. 69–92.

Smith, S. (1994) Optimization techniques on Riemannian manifolds. *Fields Inst. Commun.*, **3**, 113–135.

Sorensen, D. (1982) Newton's method with a model trust region modification. *SIAM J. Numer. Anal.*, **19**, 409–426. doi:10.1137/0719026.

Steihaug, T. (1983) The conjugate gradient method and trust regions in large scale optimization. *SIAM J. Numer. Anal.*, **20**, 626–637.

Sun, J., Qu, Q. & Wright, J. (2017a) Complete dictionary recovery over the sphere II: recovery by Riemannian trust-region method. *IEEE Trans. Info. Theory*, **63**, 885–914. doi:10.1109/TIT.2016.2632149.

Sun, J., Qu, Q. & Wright, J. (2017b) A geometric analysis of phase retrieval. *Foundations Comput. Math.* doi:10.1007/s10208–017-9365–9.

Toint, P. (1981) Towards an efficient sparsity exploiting Newton method for minimization. *Sparse Matrices and Their Uses* (I. Duff ed). Academic Press, pp. 57–88.

Townsend, J., Koep, N. & Weichwald, S. (2016) PyManopt: a Python toolbox for optimization on manifolds using automatic differentiation. *J. Mach. Learn. Res.*, **17**, 1–5.

Udriste, C. (1994) Convex functions and optimization methods on Riemannian manifolds, Mathematics and its Applications, vol. **297**. Springer, Dordrecht: Kluwer Academic Publishers. doi:10.1007/978–94-015–8390-9.

Vandereycken, B. (2013) Low-rank matrix completion by Riemannian optimization. *SIAM J. Optim.*, **23**, 1214–1236. doi:10.1137/110845768.

Vavasis, S. (1991) *Nonlinear Optimization: Complexity Issues*. Oxford: Oxford University Press.

Yang, W., Zhang, L.-H. & Song, R. (2014) Optimality conditions for the nonlinear programming problems on Riemannian manifolds. *Pacific J. Optim.*, **10**, 415–434.

Zhang, H. & Sra, S. (2016) First-order methods for geodesically convex optimization. *Conference on Learning Theory.* Curran Associates, pp. 1617–1638.

Zhang, H., Reddi, S. & Sra S. (2016) Riemannian SVRG: fast stochastic optimization on Riemannian manifolds. *Advances in Neural Information Processing Systems.* Curran Associates. pp. 4592–4600.

Appendix A. Essentials about manifolds

We give here a simplified refresher of differential geometric concepts used in the paper, restricted to Riemannian submanifolds. All concepts are illustrated with the sphere. See Absil *et al.* (2008) for a more complete discussion, including quotient manifolds.

We endow $\mathbb{R}^n$ with the classical Euclidean metric: for all $x, y \in \mathbb{R}^n$, $\langle x, y\rangle = x^{\mathrm{T}}y$. Consider the smooth map $h\colon \mathbb{R}^n \mapsto \mathbb{R}^m$ with $m \leq n$ and the constraint set

$$\mathcal{M} = \left\{x \in \mathbb{R}^n : h(x) = 0\right\}.$$

Locally around each x, this set can be linearized by differentiating the constraint. The subspace corresponding to this linearization is the kernel of the differential of h at x (Absil *et al.*, 2008, eq. (3.19)):

$$\mathrm{T}_x\mathcal{M} = \left\{\eta \in \mathbb{R}^n : \mathrm{D}h(x)[\eta] = 0\right\}.$$

If this subspace has dimension $n - m$ for all $x \in \mathcal{M}$ then $\mathcal{M}$ is a submanifold of dimension $n - m$ of $\mathbb{R}^n$ (Absil *et al.*, 2008, Prop. 3.3.3) and $\mathrm{T}_x\mathcal{M}$ is called the tangent space to $\mathcal{M}$ at x. For example, the unit sphere in $\mathbb{R}^n$ is a submanifold of dimension $n - 1$ defined by

$$\mathcal{S}^{n-1} = \left\{x \in \mathbb{R}^n : x^{\mathrm{T}}x = 1\right\},$$

and the tangent space at x is

$$\mathrm{T}_x\mathcal{S}^{n-1} = \left\{\eta \in \mathbb{R}^n : x^{\mathrm{T}}\eta = 0\right\}.$$

By endowing each tangent space with the (restricted) Euclidean metric we turn $\mathcal{M}$ into a Riemannian submanifold of the Euclidean space $\mathbb{R}^n$. (In general, the metric could be different and would depend on x; to disambiguate, one would write $\langle\cdot, \cdot\rangle_x$.) An obvious retraction for the sphere (see Definition 2.1) is to normalize:

$$\mathrm{Retr}_x(\eta) = \frac{x + \eta}{\|x + \eta\|}.$$

Being an orthogonal projection to the manifold, this is actually a second-order retraction; see Definition 3.10 and Absil & Malick (2012, Thm. 22).

The Riemannian metric leads to the notion of Riemannian gradient of a real function f defined in an open set of $\mathbb{R}^n$ containing $\mathcal{M}$.[8] The Riemannian gradient of f at x is the (unique) tangent vector $\operatorname{grad} f(x)$ at x satisfying

$$\forall\, \eta \in \mathrm{T}_x\mathcal{M}, \quad \mathrm{D}f(x)[\eta] = \lim_{t\to 0} \frac{f(x+t\eta) - f(x)}{t} = \langle \eta, \operatorname{grad} f(x)\rangle\,.$$

In this setting the Riemannian gradient is nothing but the orthogonal projection of the Euclidean (classical) gradient $\nabla f(x)$ to the tangent space. Writing $\operatorname{Proj}_x \colon \mathbb{R}^n \to \mathrm{T}_x\mathcal{M}$ for the orthogonal projector we have (Absil *et al.*, 2008, eq. (3.37))

$$\operatorname{grad} f(x) = \operatorname{Proj}_x(\nabla f(x))\,.$$

Continuing the sphere example, the orthogonal projector is $\operatorname{Proj}_x(y) = y - \left(x^{\mathrm{T}}y\right)x$, and if $f(x) = \frac{1}{2}x^{\mathrm{T}}Ax$ for some symmetric matrix A then

$$\nabla f(x) = Ax, \qquad \text{and} \qquad \operatorname{grad} f(x) = Ax - (x^{\mathrm{T}}Ax)x.$$

Notice that the critical points of f on $\mathcal{S}^{n-1}$ coincide with the unit eigenvectors of A.

We can further define a notion of Riemannian Hessian as the projected differential of the Riemannian gradient:[9]

$$\operatorname{Hess} f(x)[\eta] = \operatorname{Proj}_x\left(\mathrm{D}\left(x \mapsto \operatorname{Proj}_x \nabla f(x)\right)(x)[\eta]\right).$$

$\operatorname{Hess} f(x)$ is a linear map from $\mathrm{T}_x\mathcal{M}$ to itself, symmetric with respect to the Riemannian metric. Given a second-order retraction (Definition 3.10), it is equivalently defined by

$$\forall\, \eta \in \mathrm{T}_x\mathcal{M}, \quad \langle \eta, \operatorname{Hess} f(x)[\eta]\rangle = \frac{\mathrm{d}^2}{\mathrm{d}t^2} f(\operatorname{Retr}_x(t\eta))\bigg|_{t=0};$$

see Absil *et al.* (2008, eq. (5.35)). Continuing our sphere example,

$$\mathrm{D}\left(x \mapsto \operatorname{Proj}_x \nabla f(x)\right)(x)[\eta] = \mathrm{D}\left(x \mapsto Ax - \left(x^{\mathrm{T}}Ax\right)x\right)(x)[\eta] = A\eta - (x^{\mathrm{T}}Ax)\eta - 2(x^{\mathrm{T}}A\eta)x.$$

Projection of the latter gives the Hessian:

$$\operatorname{Hess} f(x)[\eta] = \operatorname{Proj}_x(A\eta) - (x^{\mathrm{T}}Ax)\eta.$$

---

[8] The function f need not be defined outside of $\mathcal{M}$, but this is often the case in applications and simplifies exposition.

[9] Proper definition of Riemannian Hessians requires the notion of Riemannian connections, which we omit here; see Absil *et al.* (2008, Chapter 5).

Consider the implications of a positive semidefinite Hessian (on the tangent space):

$$\begin{aligned}\operatorname{Hess} f(x) \succeq 0 &\iff \langle \eta, \operatorname{Hess} f(x)[\eta]\rangle \geq 0 \qquad \forall\, \eta \in \mathrm{T}_x\mathcal{S}^{n-1} \\ &\iff \eta^{\mathrm{T}} A\eta \geq x^{\mathrm{T}} Ax \qquad \forall\, \eta \in \mathrm{T}_x\mathcal{S}^{n-1}, \|\eta\| = 1.\end{aligned}$$

Together with first-order conditions this implies that x is a leftmost eigenvector of A.[10] This is an example of an optimization problem on a manifold for which second-order necessary optimality conditions are also sufficient. This is not the norm.

As another (very) special example consider the case $\mathcal{M} = \mathbb{R}^n$; then, $\mathrm{T}_x\mathbb{R}^n = \mathbb{R}^n$, $\operatorname{Retr}_x(\eta) = x + \eta$ is the exponential map (*a fortiori* a second-order retraction), Proj_x is the identity, $\operatorname{grad} f(x) = \nabla f(x)$ and $\operatorname{Hess} f(x) = \nabla^2 f(x)$.

Appendix B. Compact submanifolds of Euclidean spaces

In this appendix we prove Lemmas 2.4 and 3.1, showing that if f has locally Lipschitz continuous gradient or Hessian in a Euclidean space $\mathcal{E}$ (in the usual sense), and it is to be minimized over a compact submanifold of $\mathcal{E}$, then Assumptions 2.6, 3.1 and 3.2 hold.

Proof of Lemma 2.4. By assumption, ∇f is Lipschitz continuous along any line segment in $\mathcal{E}$ joining x and y in $\mathcal{M}$. Hence, there exists L such that, for all $x, y \in \mathcal{M}$,

$$|f(y) - [f(x) + \langle \nabla f(x), y - x\rangle]| \leq \frac{L}{2}\|y - x\|^2. \tag{B.1}$$

In particular this holds for all $y = \operatorname{Retr}_x(\eta)$, for any $\eta \in \mathrm{T}_x\mathcal{M}$. Writing $\operatorname{grad} f(x)$ for the Riemannian gradient of $f|_{\mathcal{M}}$ and using that $\operatorname{grad} f(x)$ is the orthogonal projection of $\nabla f(x)$ to $\mathrm{T}_x\mathcal{M}$ (Absil *et al.*, 2008, eq. (3.37)) the inner product above decomposes as

$$\begin{aligned}\langle \nabla f(x), \operatorname{Retr}_x(\eta) - x\rangle &= \langle \nabla f(x), \eta + \operatorname{Retr}_x(\eta) - x - \eta\rangle \\ &= \langle \operatorname{grad} f(x), \eta\rangle + \langle \nabla f(x), \operatorname{Retr}_x(\eta) - x - \eta\rangle\,.\end{aligned} \tag{B.2}$$

Combining (B.1) with (B.2) and using the triangle inequality yields

$$\left|f(\operatorname{Retr}_x(\eta)) - \left[f(x) + \langle \operatorname{grad} f(x), \eta\rangle\right]\right| \leq \frac{L}{2}\|\operatorname{Retr}_x(\eta) - x\|^2 + \|\nabla f(x)\|\,\|\operatorname{Retr}_x(\eta) - x - \eta\|.$$

Since $\nabla f(x)$ is continuous on the compact set $\mathcal{M}$ there exists finite G such that $\|\nabla f(x)\| \leq G$ for all $x \in \mathcal{M}$. It remains to show there exist finite constants $\alpha, \beta \geq 0$ such that, for all $x \in \mathcal{M}$ and for all $\eta \in \mathrm{T}_x\mathcal{M}$,

$$\|\operatorname{Retr}_x(\eta) - x\| \leq \alpha\|\eta\| \text{ and} \tag{B.3}$$

$$\|\operatorname{Retr}_x(\eta) - x - \eta\| \leq \beta\|\eta\|^2. \tag{B.4}$$

[10] Indeed, any $y \in \mathcal{S}^{n-1}$ can be written as $y = \alpha x + \beta\eta$ with $x^{\mathrm{T}}\eta = 0$, $\|\eta\| = 1$ and $\alpha^2 + \beta^2 = 1$; then, $y^{\mathrm{T}}Ay = \alpha^2 x^{\mathrm{T}}Ax + \beta^2\eta^{\mathrm{T}}A\eta + 2\alpha\beta\eta^{\mathrm{T}}Ax$; by first-order condition, $\eta^{\mathrm{T}}Ax = (x^{\mathrm{T}}Ax)\eta^{\mathrm{T}}x = 0$, and by second-order condition, $y^{\mathrm{T}}Ay \geq (\alpha^2 + \beta^2)x^{\mathrm{T}}Ax = x^{\mathrm{T}}Ax$; hence $x^{\mathrm{T}}Ax$ is minimal over $\mathcal{S}^{n-1}$.

For small η, this will follow from $\mathrm{Retr}_x(\eta) = x + \eta + \mathcal{O}\left(\|\eta\|^2\right)$ by Definition 2.1; for large η this will follow *a fortiori* from compactness. This will be sufficient to conclude, as then we will have for all $x \in \mathcal{M}$ and $\eta \in \mathrm{T}_x\mathcal{M}$ that

$$\left|f(\mathrm{Retr}_x(\eta)) - \left[f(x) + \langle \mathrm{grad}\, f(x), \eta\rangle\right]\right| \leq \left(\frac{L}{2}\alpha^2 + G\beta\right)\|\eta\|^2.$$

More formally our assumption that the retraction is defined and smooth over the whole tangent bundle *a fortiori* ensures the existence of $r > 0$ such that Retr is smooth on $K = \{\eta \in \mathrm{T}\mathcal{M} : \|\eta\| \leq r\}$, a compact subset of the tangent bundle (K consists of a ball in each tangent space). First, we determine α; see (B.3). For all $\eta \in K$ we have

$$\begin{aligned}\|\mathrm{Retr}_x(\eta) - x\| &\leq \int_0^1 \left\|\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Retr}_x(t\eta)\right\| \mathrm{d}t = \int_0^1 \|\mathrm{DRetr}_x(t\eta)[\eta]\|\, \mathrm{d}t \\ &\leq \int_0^1 \max_{\xi\in K}\|\mathrm{DRetr}(\xi)\|\|\eta\|\, \mathrm{d}t = \max_{\xi\in K}\|\mathrm{DRetr}(\xi)\|\|\eta\|,\end{aligned}$$

where the max exists and is finite owing to compactness of K and smoothness of Retr on K; note that this is uniform over both x and η. (If $\xi \in \mathrm{T}_z\mathcal{M}$ the notation $\mathrm{DRetr}(\xi)$ refers to $\mathrm{DRetr}_z(\xi)$.) For all $\eta \notin K$ we have

$$\|\mathrm{Retr}_x(\eta) - x\| \leq \mathrm{diam}(\mathcal{M}) \leq \frac{\mathrm{diam}(\mathcal{M})}{r}\|\eta\|,$$

where $\mathrm{diam}(\mathcal{M})$ is the maximal distance between any two points on $\mathcal{M}$: finite by compactness of $\mathcal{M}$. Combining, we find that (B.3) holds with

$$\alpha = \max\left(\max_{\xi\in K}\|\mathrm{DRetr}(\xi)\|, \frac{\mathrm{diam}(\mathcal{M})}{r}\right).$$

Inequality (B.4) is established along similar lines. For all $\eta \in K$ we have

$$\begin{aligned}\|\mathrm{Retr}_x(\eta) - x - \eta\| &\leq \int_0^1 \left\|\frac{\mathrm{d}}{\mathrm{d}t}(\mathrm{Retr}_x(t\eta) - x - t\eta)\right\| \mathrm{d}t = \int_0^1 \|\mathrm{DRetr}_x(t\eta)[\eta] - \eta\|\, \mathrm{d}t \\ &\leq \int_0^1 \|\mathrm{DRetr}_x(t\eta) - \mathrm{Id}\,\|\|\eta\|\, \mathrm{d}t \leq \frac{1}{2}\max_{\xi\in K}\|\mathrm{D}^2\mathrm{Retr}(\xi)\|\|\eta\|^2,\end{aligned}$$

where the last inequality follows from $\mathrm{DRetr}_x(0_x) = \mathrm{Id}$ and

$$\|\mathrm{DRetr}_x(t\eta) - \mathrm{Id}\,\| \leq \int_0^1 \left\|\frac{\mathrm{d}}{\mathrm{d}s}\mathrm{DRetr}_x(st\eta)\right\| \mathrm{d}s \leq \|t\eta\|\int_0^1 \left\|\mathrm{D}^2\mathrm{Retr}_x(t\eta)\right\| \mathrm{d}s.$$

The case $\eta \notin K$ is treated as before:

$$\|\mathrm{Retr}_x(\eta) - x - \eta\| \leq \|\mathrm{Retr}_x(\eta) - x\| + \|\eta\| \leq \frac{\mathrm{diam}(\mathcal{M}) + r}{r^2}\|\eta\|^2.$$

Combining, we find that (B.4) holds with

$$\beta = \max\left(\frac{1}{2}\max_{\xi \in K}\left\|\mathrm{D}^2\mathrm{Retr}(\xi)\right\|, \frac{\mathrm{diam}(\mathcal{M}) + r}{r^2}\right),$$

which concludes the proof. □

We now prove the corresponding second-order result, whose aim is to verify Assumptions 3.2.

Proof of Lemma 3.1. By assumption $\nabla^2 f$ is Lipschitz continuous along any line segment in $\mathcal{E}$ joining x and y in $\mathcal{M}$. Hence, there exists L such that, for all $x, y \in \mathcal{M}$,

$$\left|f(y) - \left[f(x) + \langle \nabla f(x), y - x\rangle + \tfrac{1}{2}\left\langle y - x, \nabla^2 f(x)[y - x]\right\rangle\right]\right| \leq \frac{L}{6}\|y - x\|^3. \tag{B.5}$$

Fix $x \in \mathcal{M}$. Let Proj_x denote the orthogonal projector from $\mathcal{E}$ to $\mathrm{T}_x\mathcal{M}$. Let $\mathrm{grad}\, f(x)$ be the Riemannian gradient of $f|_{\mathcal{M}}$ at x and let $\mathrm{Hess}\, f(x)$ be the Riemannian Hessian of $f|_{\mathcal{M}}$ at x (a symmetric operator on $\mathrm{T}_x\mathcal{M}$). For Riemannian submanifolds of Euclidean spaces we have these explicit expressions with $\eta \in \mathrm{T}_x\mathcal{M}$—see Absil *et al.* (2008, eqs. (3.37), (5.15), Def. (5.5.1)) and Absil *et al.* (2013):

$$\begin{aligned}
\mathrm{grad}\, f(x) &= \mathrm{Proj}_x \nabla f(x), \text{ and} \\
\langle \eta, \mathrm{Hess}\, f(x)[\eta]\rangle &= \left\langle \eta, \mathrm{D}\left(x \mapsto \mathrm{Proj}_x \nabla f(x)\right)(x)[\eta]\right\rangle \\
&= \left\langle \eta, \left(\mathrm{D}\left(x \mapsto \mathrm{Proj}_x\right)(x)[\eta]\right)[\nabla f(x)] + \mathrm{Proj}_x \nabla^2 f(x)[\eta]\right\rangle \\
&= \langle II(\eta, \eta), \nabla f(x)\rangle + \left\langle \eta, \nabla^2 f(x)[\eta]\right\rangle,
\end{aligned}$$

where II, as implicitly defined above, is the *second fundamental form* of $\mathcal{M}$: $II(\eta, \eta)$ is a normal vector to the tangent space at x, capturing the second-order geometry of $\mathcal{M}$—see Absil *et al.* (2009, 2013) and Monera *et al.* (2014) for presentations relevant to our setting. In particular, $II(\eta, \eta)$ is the acceleration in $\mathcal{E}$ at x of a geodesic $\gamma(t)$ on $\mathcal{M}$ defined by $\gamma(0) = x$ and $\dot\gamma(0) = \eta$: $\ddot\gamma(0) = II(\eta, \eta)$ (O'Neill, 1983, Cor. 4.9).

Let $\eta \in \mathrm{T}_x\mathcal{M}$ be arbitrary; $y = \mathrm{Retr}_x(\eta) \in \mathcal{M}$. Then

$$\begin{aligned}
\langle \nabla f(x), y - x\rangle - \langle \mathrm{grad}\, f(x), \eta\rangle &= \langle \nabla f(x), y - x - \eta\rangle \text{ and} \\
\left\langle y - x, \nabla^2 f(x)[y - x]\right\rangle - \langle \eta, \mathrm{Hess}\, f(x)[\eta]\rangle &= 2\left\langle \eta, \nabla^2 f(x)[y - x - \eta]\right\rangle \\
&\quad + \left\langle y - x - \eta, \nabla^2 f(x)[y - x - \eta]\right\rangle \\
&\quad - \langle \nabla f(x), II(\eta, \eta)\rangle\,.
\end{aligned}$$

Since $\mathcal{M}$ is compact and f is twice continuously differentiable, there exist G, H, independent of x, such that $\|\nabla f(x)\| \leq G$ and $\left\|\nabla^2 f(x)\right\| \leq H$ (the latter is the induced operator norm). Combining with (B.5) and using the triangle and Cauchy–Schwarz inequalities multiple times,

$$\left| f(y) - \left[f(x) + \langle \operatorname{grad} f(x), \eta \rangle + \frac{1}{2} \langle \eta, \operatorname{Hess} f(x)[\eta] \rangle \right] \right| \\ \leq \frac{L}{6}\|y - x\|^3 + G\left\| y - x - \eta - \frac{1}{2} II(\eta, \eta) \right\| + H\|\eta\|\|y - x - \eta\| + \frac{1}{2} H\|y - x - \eta\|^2.$$

Using the same argument as for Lemma 2.4 we can find finite constants α, β independent of x and η such that (B.3) and (B.4) hold. Use $\|y - x - \eta\|^2 \leq \|y - x - \eta\| (\|y - x\| + \|\eta\|) \leq \beta(\alpha + 1)\|\eta\|^3$ to upper bound the right-hand side above with

$$\left(\frac{L}{6}\alpha^3 + H\beta + \frac{H\beta(\alpha + 1)}{2} \right) \|\eta\|^3 + G\left\| y - x - \eta - \frac{1}{2} II(\eta, \eta) \right\|.$$

We turn to the last term. Consider $K \subset \mathrm{T}\mathcal{M}$ as defined in the proof of Lemma 2.4 for some $r > 0$. If $\eta \notin K$, that is, $\|\eta\| > r$, then since II is bilinear for a fixed $x \in \mathcal{M}$, we can define

$$\|II\| = \max_{x \in \mathcal{M}} \max_{\xi \in \mathrm{T}_x\mathcal{M}, \|\xi\| \leq 1} \|II(\xi, \xi)\|$$

(finite by continuity and compactness) so that $\|II(\eta, \eta)\| \leq \|II\| \|\eta\|^2$. Then,

$$\left\| y - x - \eta - \frac{1}{2} II(\eta, \eta) \right\| \leq \|y - x\| + \|\eta\| + \frac{1}{2}\|II(\eta, \eta)\| \leq \left(\frac{\operatorname{diam}(\mathcal{M})}{r^3} + \frac{1}{r^2} + \frac{1}{2}\frac{\|II\|}{r} \right) \|\eta\|^3.$$

Now assume $\eta \in K$, that is, $\|\eta\| \leq r$. Consider $\phi(t) = \operatorname{Retr}_x(t\eta)$ (a curve on $\mathcal{M}$) and let ϕ'' denote its acceleration on $\mathcal{M}$ and $\ddot{\phi}$ denote its acceleration in $\mathcal{E}$, while $\dot{\phi} = \phi'$ denotes velocity along the curve. It holds that $\ddot{\phi}(t) = \phi''(t) + II(\dot{\phi}(t), \dot{\phi}(t))$ (O'Neill, 1983, Cor. 4.9). Since Retr is a second-order retraction, acceleration on $\mathcal{M}$ is zero at $t = 0$, that is, $\phi''(0) = 0$, so that $\phi(0) = x$, $\dot{\phi}(0) = \eta$ and $\ddot{\phi}(0) = II(\eta, \eta)$. Then by Taylor expansion of ϕ in $\mathcal{E}$,

$$y = \operatorname{Retr}_x(\eta) = \phi(1) = x + \eta + \frac{1}{2} II(\eta, \eta) + R_3(\eta),$$

where

$$\|R_3(\eta)\| = \left\| \int_0^1 \frac{(1 - t)^2}{2} \dddot{\phi}(t)\, \mathrm{d}t \right\| \leq \frac{1}{6} \max_{\xi \in K} \left\| \mathrm{D}^3 \operatorname{Retr}(\xi) \right\| \|\eta\|^3.$$

The combined arguments ensure existence of a constant γ, independent of x and η, such that

$$\left\| y - x - \eta - \frac{1}{2} II(\eta, \eta) \right\| \leq \gamma \|\eta\|^3.$$

Combining, we find that for all $x \in \mathcal{M}$ and $\eta \in \mathrm{T}_x\mathcal{M}$,

$$\left| f(\mathrm{Retr}_x(\eta)) - \left[f(x) + \langle \mathrm{grad}\, f(x), \eta \rangle + \frac{1}{2} \langle \eta, \mathrm{Hess}\, f(x)[\eta] \rangle \right] \right| \leq \left(\frac{L}{6}\alpha^3 + \frac{H\beta(\alpha+3)}{2} + \gamma \right) \|\eta\|^3.$$

Since Retr is a second-order retraction, $\mathrm{Hess}\, f(x)$ coincides with the Hessian of the pullback $f \circ \mathrm{Retr}_x$ (Lemma 3.9). This establishes Assumption 3.2. □

Appendix C. Proof of Lemma 2.7 about Armijo line-search

Proof of Lemma 2.7. By Assumption 2.6, upper bound (2.5) holds with $\eta = t\eta_k^0$ for any t such that $\|\eta\| \leq \varrho_k$:

$$f(x_k) - f\left(\mathrm{Retr}_{x_k}\left(t \cdot \eta_k^0\right)\right) \geq t\left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle - \frac{Lt^2}{2}\left\|\eta_k^0\right\|^2. \tag{C.1}$$

We determine a sufficient condition on t for the stopping criterion in Algorithm 2 to trigger. To this end observe that the right-hand side of (C.1) dominates $c_1 t\left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle$ if

$$t(1-c_1) \cdot \left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle \geq \frac{Lt^2}{2}\left\|\eta_k^0\right\|^2.$$

Thus, the stopping criterion in Algorithm 2 is satisfied in particular for all t in

$$\left[0, \frac{2(1-c_1)\left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle}{L_g \|\eta_k^0\|^2}\right] \supseteq \left[0, \frac{2c_2(1-c_1)\|\mathrm{grad}\, f(x_k)\|}{L_g\|\eta_k^0\|}\right] \supseteq \left[0, \frac{2c_2(1-c_1)}{c_4 L_g}\right].$$

Unless it equals $\bar{t}_k$, the returned t cannot be smaller than τ times the last upper bound. In all cases the cost decrease satisfies

$$\begin{aligned} f(x_k) - f\left(\mathrm{Retr}_{x_k}\left(t \cdot \eta_k^0\right)\right) &\geq c_1 t\left\langle -\mathrm{grad}\, f(x_k), \eta_k^0 \right\rangle \\ &\geq c_1 c_2 t \|\mathrm{grad}\, f(x_k)\| \left\|\eta_k^0\right\| \\ &\geq c_1 c_2 c_3 t \|\mathrm{grad}\, f(x_k)\|^2. \end{aligned}$$

To count the number of iterations consider that checking whether $t = \bar{t}_k$ satisfies the stopping criterion requires one cost evaluation. Following that, t is reduced by a factor τ exactly $\log_\tau(t/\bar{t}_k) = \log_{\tau^{-1}}(\bar{t}_k/t)$ times, each followed by one cost evaluation. □

Appendix D. Proofs for Section 3.5 about H_k and the Hessians

Proof of Lemma 3.9. The Hessian of f and that of the pullback are related by the following formulas. See (Absil *et al.*, 2008, Chapter 5) for the precise meanings of the differential operators D and d. For all η in $\mathrm{T}_x\mathcal{M}$, writing $\hat{f}_x = f \circ \mathrm{Retr}_x$ for convenience,

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} f(\mathrm{Retr}_x(t\eta)) &= \left\langle \mathrm{grad}\, f(\mathrm{Retr}_x(t\eta)), \frac{\mathrm{D}}{\mathrm{d}t}\mathrm{Retr}_x(t\eta) \right\rangle, \\
\left\langle \nabla^2 \hat{f}_x(0_x)[\eta], \eta \right\rangle &= \frac{\mathrm{d}^2}{\mathrm{d}t^2} f(\mathrm{Retr}_x(t\eta))\Big|_{t=0} \\
&= \left\langle \mathrm{Hess}\, f(x)\,[\mathrm{DRetr}_x(0_x)[\eta]], \frac{\mathrm{D}}{\mathrm{d}t}\mathrm{Retr}_x(t\eta)\Big|_{t=0} \right\rangle \\
&\quad + \left\langle \mathrm{grad}\, f(x), \frac{\mathrm{D}^2}{\mathrm{d}t^2}\mathrm{Retr}_x(t\eta)\Big|_{t=0} \right\rangle \\
&= \langle \mathrm{Hess}\, f(x)[\eta], \eta \rangle + \left\langle \mathrm{grad}\, f(x), \frac{\mathrm{D}^2}{\mathrm{d}t^2}\mathrm{Retr}_x(t\eta)\Big|_{t=0} \right\rangle.
\end{aligned}$$

(To get the third equality it is assumed one is working with the Levi-Civita connection, so that Hess f is indeed the Riemannian Hessian.) Since the acceleration of the retraction is bounded, we get the result via Cauchy–Schwarz. □

Proof of Proposition 3.11. Combine $\|\mathrm{grad}\, f(x_k)\| \leq \varepsilon_g$ and $H_k \succcurlyeq -\varepsilon_H \,\mathrm{Id}$ with

$$\left\| \mathrm{Hess}\, f(x_k) - \nabla^2 \hat{f}_{x_k}(0_{x_k}) \right\| \leq a_k \cdot \|\mathrm{grad}\, f(x_k)\| \quad \text{and} \quad \left\| \nabla^2 \hat{f}_k(0_{x_k}) - H_k \right\| \leq \delta_k$$

by the triangular inequality. □

Appendix E. Complexity dependence on n in the Max-Cut example

This appendix supports Section 4. By Proposition 4.1, running Algorithm 3 with $\varepsilon_g = \infty$ and $\varepsilon_H = \frac{2\delta}{n}$ yields a solution Y within a gap δ from the optimal value of (4.2). Let $\underline{f}$ and $\overline{f}$ denote the minimal and maximal values of $f(Y) = \langle C, YY^{\mathrm{T}} \rangle$ over $\mathcal{M}$ (see (4.3)), respectively, with metric $\langle A, B \rangle = \mathrm{Tr}\left(A^{\mathrm{T}}B\right)$ and associated Frobenius norm $\|\cdot\|_F$. Then using $\rho' = 1/10$, setting $c_3 = 1/2$ in Assumption 3.8 as allowed by Lemma 3.4 and using the true Hessian of the pullbacks for H_k so that $c_1 = 0$ in Assumption 3.5, Theorem 3.4 guarantees that Algorithm 3 returns an answer in at most

$$214(\overline{f} - \underline{f}) \cdot L_H^2 \cdot \frac{1}{\varepsilon_H^3} + \log \text{ term} \tag{E.1}$$

iterations. Using the LDL^{T}-factorization strategy of Lemma 3.3 with a randomly generated orthonormal basis at each tangent space encountered, since $\dim \mathcal{M} = n^2$ for $p = n + 1$, the cost of each iteration is $\mathcal{O}\left(n^6\right)$ arithmetic operations (dominated by the cost of the LDL^{T} factorization). It remains to bound L_H, in compliance with Assumption 3.2.

Let $g\colon \mathbb{R} \to \mathbb{R}$ be defined as $g(t) = f(\mathrm{Retr}_Y(t\dot{Y}))$. Then using a Taylor expansion,

$$f(\mathrm{Retr}_Y(\dot{Y})) = g(1) = g(0) + g'(0) + \frac{1}{2}g''(0) + \frac{1}{6}g'''(t) \tag{E.2}$$

for some $t \in (0, 1)$. Let $\hat{f}_Y = f \circ \mathrm{Retr}_Y$. Definition 2.1 for retractions implies

$$g(0) = f(Y), \qquad g'(0) = \langle \mathrm{grad}\, f(Y), \dot{Y} \rangle, \qquad g''(0) = \left\langle \dot{Y}, \nabla^2 \hat{f}_Y(0_Y)[\dot{Y}] \right\rangle, \tag{E.3}$$

so that it only remains to bound $|g'''(t)|$ uniformly over $Y, \dot{Y}$ and $t \in [0, 1]$.

For this example it is easier to handle g''' if the retraction used is the exponential map (similar bounds can be obtained with the orthogonal projection retraction; see Mei *et al.*, 2017, Lemmas 4 and 5). This map is known in explicit form and is cheap to compute for the sphere $\mathbb{S}^n = \{x \in \mathbb{R}^{n+1} : x^{\mathrm{T}}x = 1\}$. Indeed, if $x \in \mathbb{S}^n$ and $\eta \in \mathrm{T}_x\mathbb{S}^n$, following Absil *et al.* (2008, Ex. 5.4.1),

$$\gamma(t) = \mathrm{Exp}_x(t\eta) = \cos(t\|\eta\|)x + \sin(t\|\eta\|)\frac{1}{\|\eta\|}\eta. \tag{E.4}$$

Conceiving of γ as a map from $\mathbb{R}$ to $\mathbb{R}^{n+1}$ its differentials are easily derived:

$$\dot{\gamma}(t) = -\|\eta\| \sin(t\|\eta\|)x + \cos(t\|\eta\|)\eta, \qquad \ddot{\gamma}(t) = -\|\eta\|^2\gamma(t), \qquad \dddot{\gamma}(t) = -\|\eta\|^2\dot{\gamma}(t). \tag{E.5}$$

Extending this map rowwise gives the exponential map for $\mathcal{M}$—of course, this is a second-order retraction. We define $\Phi(t) = \mathrm{Retr}_Y(t\dot{Y})$ and $g(t) = f(\mathrm{Retr}_Y(t\dot{Y})) = \langle C\Phi(t), \Phi(t)\rangle$. In particular $\ddot{\Phi}(t) = -D\Phi(t)$ and $\dddot{\Phi}(t) = -D\dot{\Phi}(t)$, where $D = \mathrm{diag}\left(\|\dot{y}_1\|^2, \ldots, \|\dot{y}_n\|^2\right)$ and $\dot{y}_k^{\mathrm{T}}$ is the kth row of $\dot{Y}$. As a result, for a given Y and $\dot{Y}$, a little bit of calculus gives

$$g'''(t) = -6\left\langle C\dot{\Phi}(t), D\Phi(t)\right\rangle - 2\left\langle C\Phi(t), D\dot{\Phi}(t)\right\rangle. \tag{E.6}$$

Using Cauchy–Schwarz multiple times, as well as the inequality $\|AB\|_{\mathrm{F}} \leq \|A\|_2 \|B\|_{\mathrm{F}}$ where $\|A\|_2$ denotes the largest singular value of A, and using that $\|\Phi(t)\|_{\mathrm{F}} = \sqrt{n}$ and $\|\dot{\Phi}(t)\|_{\mathrm{F}} = \|\dot{Y}\|_{\mathrm{F}}$ for all t, and additionally that $\|D\|_2 \leq \mathrm{Tr}(D) = \|\dot{Y}\|_{\mathrm{F}}^2$, it follows that

$$\sup_{Y \in \mathcal{M}, \dot{Y} \in \mathrm{T}_Y\mathcal{M}, \dot{Y} \neq 0, t \in (0,1)} \frac{|g'''(t)|}{\|\dot{Y}\|_{\mathrm{F}}^3} \leq 8\,\|C\|_2\sqrt{n}. \tag{E.7}$$

As a result an acceptable constant L_H for Assumption 3.2 is $L_H = 8\,\|C\|_2\sqrt{n}$.

Combining all the statements of this section it follows that a solution Y within an absolute gap δ of the optimal value can be obtained for problem (4.2) using Algorithm 3 in at most $\mathcal{O}\left((\bar{f} - \underline{f})\,\|C\|_2^2 \cdot n^{10} \cdot \frac{1}{\delta^3}\right)$ arithmetic operations, neglecting the additive logarithmic term.

Note that, following Mei *et al.* (2017, Appendix A.2, points 1 and 2), it is also possible to bound L_H as $6\,\|C\|_2 + 2\|C\|_1$, where $\|\cdot\|_1$ is the ℓ_1 operator norm. This reduces the explicit dependence on n from n^{10} to n^9 in the bound on the total amount of work.