SIAM J. MATRIX ANAL. APPL.
Vol. 39, No. 3, pp. 1181–1207
© 2018 Society for Industrial and Applied Mathematics

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

ADAPTIVE QUADRATICALLY REGULARIZED NEWTON METHOD FOR RIEMANNIAN OPTIMIZATION*

JIANG HU†, ANDRE MILZAREK†, ZAIWEN WEN†, AND YAXIANG YUAN‡

Abstract. Optimization on Riemannian manifolds widely arises in eigenvalue computation, density functional theory, Bose–Einstein condensates, low rank nearest correlation, image registration, signal processing, and so on. We propose an adaptive quadratically regularized Newton method which approximates the original objective function by the second-order Taylor expansion in Euclidean space but keeps the Riemannian manifold constraints. The regularization term in the objective function of the subproblem enables us to utilize a Cauchy-point–like condition as the standard trust-region method for proving global convergence. The subproblem can be solved inexactly either by first-order methods or by performing corresponding Riemannian Newton–type steps. In the latter case, we can further take advantage of negative curvature directions. Both global convergence and superlinear local convergence are guaranteed under mild conditions. Extensive computational experiments and comparisons with other state-of-the-art methods indicate that the proposed algorithm is very promising.

Key words. Riemannian optimization, regularization, trust-region methods, inexactness

AMS subject classifications. 15A18, 65K10, 65F15, 90C26, 90C30

DOI. 10.1137/17M1142478

1. Introduction.

We consider minimization problems on a Riemannian manifold of the form

$$\min_{x\in\mathcal{M}} \quad f(x), \tag{1.1}$$

where $\mathcal{M}$ is a Riemannian submanifold of a Euclidean space $\mathcal{E}$ and $f : \mathcal{M} \to \mathbb{R}$ is a smooth real-valued function on $\mathcal{M}$. This problem widely exists in eigenvalue decomposition [2], density functional theory [38], Bose–Einstein condensates [40], low rank nearest correlation matrix completion [37], and many other varieties of applications.

Riemannian optimization has been extensively studied over decades. Since problem (1.1) can be viewed as a general nonlinear optimization problem with constraints, many standard algorithms [29] can be applied to it directly. These algorithms may not be efficient since they do not utilize the intrinsic structure of the manifold. A first and basic class of manifold optimization methods can be obtained via modifying and transferring the nonlinear programming approaches to the manifold setting. In particular, by performing a curvilinear search along the geodesic, Gabay [16], Udrişte [35], Yang [43], and Smith [33] propose globally convergent steepest descent, Newton, quasi-Newton, and conjugate gradient methods, respectively. Because the computation of

*Received by the editors August 7, 2017; accepted for publication (in revised form) by P.-A. Absil April 26, 2018; published electronically July 12, 2018.
http://www.siam.org/journals/simax/39-3/M114247.html
Funding: The research of the first author was supported in part by the Elite Program of Computational and Applied Mathematics for Ph.D. Candidates at Peking University. The research of the second author was supported in part by the Boya Postdoctoral Fellowship. The research of the third author was supported in part by NSFC grants 11421101 and 91730302, and by the National Basic Research Project under grant 2015CB856002. The research of the fourth author was supported in part by NSFC grants 11331012 and 11461161005.

†Beijing International Center for Mathematical Research, Peking University, Beijing, China (jianghu@pku.edu.cn, andremilzarek@bicmr.pku.edu.cn, wenzw@pku.edu.cn).
‡State Key Laboratory of Scientific and Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China (yyx@lsec.cc.ac.cn).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

the geodesic may be difficult and expensive, Absil, Mahony, and Sepulchre [2] and Absil and Malick [4] develop a first-order approximation called retraction to the geodesic. The previously mentioned algorithms can be generalized by replacing the geodesic by the retraction, and their global and local convergence properties have been analyzed in [1, 2]. Qi [31] and Huang, Absil, and Gallivan [19, 20] and Huang, Gallivan, and Absil [22] propose an extensive class of quasi-Newton methods for Riemannian manifold problems based on retractions and vector transport. In [2], a nonlinear conjugate gradient method for Riemannian manifold problems is presented. Vandereycken [37] and Kressner, Steinlechner, and Vandereycken [26] show that algorithms using the geometry of a manifold can be efficient on a large variety of applications. Boumal, Absil, and Cartis [7] establish global convergence rates for optimization methods on manifolds. Moreover, a selection of Riemannian first-order and second-order methods has been implemented in the software package Manopt [8].

Optimization over the Stiefel manifold, i.e., problems with orthogonality constraints, is an important special case of Riemannian optimization. Edelman, Arias, and Smith [15] analyze the geometry of this specific manifold and propose Newton and conjugate gradient methods along the geodesic. Wen and Yin [39] propose a constraint-preserving algorithm on the Stiefel manifold. Other related first-order methods are presented in [45, 24, 17]. Wen et al. [38] develop an adaptively regularized Newton method which uses a quadratic approximation with exact Euclidean Hessian of the original problem. It often exhibits a superlinear or quadratic local convergence rate when the subproblem is solved accurately. This method has also been extended to Bose–Einstein condensates in [40].

In this paper, we extend the regularized Newton method in [38, 40] to general Riemannian optimization problems. Specifically, we approximate problem (1.1) and construct a subproblem by adding a regularization term to the second-order Taylor expansion of the objective function in Euclidean space. This leads to a class of Euclidean-based model problems that is generally different from classical trust-region–type approaches on Riemannian manifolds [2]. Typically, the resulting subproblems are easier to solve than the original problem to a certain extent. We show that, whenever the subproblem can be handled efficiently, a fast rate of convergence can be achieved. Since a regularization term is added, global convergence can be ensured by adjusting the regularization parameters appropriately. In fact, convergence can be guaranteed even if the subproblem is only solved inexactly as long as it attains a reduction similar to that of a single gradient descent step. Different from minimizing the subproblem by the gradient-type methods in [38, 40], we develop a modified Newton method using the conjugate gradient method to solve the Newton equation followed by a curvilinear search. In particular, our algorithm detects directions of negative curvature. We combine them with the previous conjugate directions to construct new search directions and update the regularization parameter based on the negative curvature information. Our extensive numerical experiments show that the proposed method is promising and performs at least comparably well.

We should point out that similar second-order–type methods have also been developed for composite convex programs where the objective function is a summation of a smooth function and an ℓ_1-norm or more general convex term. The subproblem in the proximal Newton method by Lee, Sun, and Saunders [28] keeps the ℓ_1-norm function but approximates the smooth part by its second-order Taylor expansion. A first-order method is then used to solve the resulting proximal subproblem. Byrd, Nocedal, and Oztoprak [9] essentially consider the same algorithm but propose a specialized active set strategy to solve the quadratic subproblem. Recently, there also has been

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

some progress on regularization methods for general nonconvex optimization problems in the Euclidean space. For instance, Karas, Santos, and Svaiter [25] consider a quadratically regularized scheme and propose algebraic rules to update the regularization parameter via utilizing the Lipschitz constant and the smallest eigenvalue of the Hessian. In [6], a cubic sufficient-descent condition is incorporated to further improve existing worst-case complexity results. Moreover, Agarwal et al. [5] develop a fast conjugate gradient–type method for cubic regularization that uses the specific properties of the subproblem. Several other and related methods that exploit negative curvature and acceleration mechanisms are presented in [30, 13, 10]. Our approach is based on similar ideas and directions, but it generalizes these techniques to the manifold setting.

This paper is organized as follows. In section 2, we review some preliminaries on Riemannian optimization and present the Riemannian gradient method. The adaptive regularized Newton method is proposed in section 3, and its convergence properties are analyzed in section 4. Finally, robustness and efficiency of the proposed algorithms are demonstrated based on several practical examples in section 5.

1.1. Notation. Let $(\mathcal{M}, g)$ be a Riemannian manifold. By $\Im_x(\mathcal{M})$, we denote the set of all real-valued functions f defined in a neighborhood of x in $\mathcal{M}$. For a given differentiable function f and a point $x \in \mathcal{M}$, $\nabla f(x)$ ($\nabla^2 f(x)$) and $\operatorname{grad} f(x)$ ($\operatorname{Hess} f(x)$) denote the Euclidean and Riemannian gradient (Hessian) of f, respectively. Let $\langle \cdot, \cdot \rangle$ ($\| \cdot \|$) and $\langle \cdot, \cdot \rangle_x$ ($\| \cdot \|_x$) be the inner product (norm) with Euclidean and Riemannian metric, respectively.

2. Preliminaries on Riemannian optimization. Many concepts of Riemannian optimization can be regarded as generalizations of the theory and algorithms from unconstrained Euclidean optimization to problems on manifolds. A detailed description of the properties of a few commonly used manifold algorithms are given in [2]. Here, we only introduce some necessary definitions briefly.

A d-dimensional manifold $\mathcal{M}$ is a Hausdorff and second-countable topological space, which is homeomorphic to the d-dimensional Euclidean space locally via a family of charts. When the transition maps of intersecting charts are smooth, manifold $\mathcal{M}$ is called a smooth manifold. A function f on $\mathcal{M}$ is said to be C^k at a point x if $f \circ \psi^{-1} : \psi(U) \subset \mathbb{R}^d \to \mathbb{R}$ is C^k, in which U is an open set in $\mathcal{M}$ containing x and ψ is the mapping defining the chart. A tangent vector ξ_x to $\mathcal{M}$ at x is a mapping such that there exists a curve γ on $\mathcal{M}$ with $\gamma(0) = x$, satisfying

$$\xi_x u := \dot{\gamma}(0)u \triangleq \left. \frac{\mathrm{d}(u(\gamma(t)))}{\mathrm{d}t} \right|_{t=0} \quad \forall\, u \in \Im_x(\mathcal{M}).$$

Then the tangent space $\mathcal{T}_x\mathcal{M}$ to $\mathcal{M}$ is defined as the set of all tangent vectors to $\mathcal{M}$ at x. If the manifold $\mathcal{M}$ can be equipped with a smoothly varying inner product $\langle \cdot, \cdot \rangle_x$ between the tangent vectors of the same tangent space, then $\mathcal{M}$ is called a Riemannian manifold. Here, we will always assume that $\mathcal{M}$ is a Riemannian submanifold of a Euclidean space $\mathcal{E}$; see, e.g., [2, section 3.6] for further details. The norm induced by the Riemannian metric is equivalent to the Euclidean norm; i.e., for all $x \in \mathcal{M}$ there exist parameters $\varpi_x^m, \varpi_x^M > 0$, which depend continuously on x such that

$$\varpi_x^m \|\xi\|_x^2 \le \|\xi\|^2 \le \varpi_x^M \|\xi\|_x^2 \quad \forall\, \xi \in \mathcal{T}_x\mathcal{M}. \tag{2.1}$$

The gradient of a real-valued function f on the Riemannian manifold is defined as the unique tangent vector satisfying $\langle \operatorname{grad} f(x), \xi \rangle_x = Df(x)[\xi]$ for all $\xi \in \mathcal{T}_x\mathcal{M}$, where $Df(x)[\xi] = \xi_x f$ and $\operatorname{grad} f(x)$ is called the Riemannian gradient of f at x.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

The Riemannian Hessian of f is a linear mapping from $\mathcal{T}_x\mathcal{M}$ to $\mathcal{T}_x\mathcal{M}$ defined by $\mathrm{Hess}\, f(x)[\xi] = \tilde{\nabla}_\xi \mathrm{grad}\, f(x)$ for all $\xi \in \mathcal{T}_x\mathcal{M}$, where $\tilde{\nabla}$ is the Riemannian connection which is a unique symmetric affine connection satisfying the Levi–Civita conditions [1]. We refer to [2] for a more detailed discussion of the Riemannian gradient and Hessian.

First- and second-order optimality conditions for Riemannian optimization problems take a similar form as standard optimality conditions in the Euclidean space. In particular, let $\mathcal{M}$ be a smooth manifold, and let $f : \mathcal{M} \to \mathbb{R}$ be a smooth function on $\mathcal{M}$. Suppose that $x_* \in \mathcal{M}$ is a *stationary point* of problem (1.1); i.e., it holds that $\mathrm{grad}\, f(x_*) = 0$. Furthermore, let $\mathrm{Hess}\, f(x_*)$ be positive definite on $\mathcal{T}_{x_*}\mathcal{M}$ (w.r.t. the Riemannian metric); then by [42, Corollary 4.3], x_* is a strict local solution of (1.1). Analogous second-order necessary conditions are presented in [42].

2.1. Gradient-type methods on manifold. We next describe a few gradient-type methods for solving (1.1). These methods generalize the concept of backtracking line search and gradient descent to the manifold setting and are based on so-called retractions. A retraction R on $\mathcal{M}$ is a smooth mapping from the tangent bundle $\mathcal{TM} := \bigcup_{x\in\mathcal{M}} \mathcal{T}_x\mathcal{M}$ to the manifold $\mathcal{M}$. Moreover, the restriction R_x of R to $\mathcal{T}_x\mathcal{M}$ has to satisfy $R_x(0_x) = x$ and $\mathrm{D}R_x(0_x) = \mathrm{id}_{\mathcal{T}_x\mathcal{M}}$, where $\mathrm{id}_{\mathcal{T}_x\mathcal{M}}$ is the identity mapping on $\mathcal{T}_x\mathcal{M}$.

Given a retraction R, the curvilinear search method computes

$$x_{k+1} = R_{x_k}(t_k\eta_k),$$

where $\eta_k \in \mathcal{T}_{x_k}\mathcal{M}$ and t_k is a scalar. Similar to Euclidean line search methods, η_k is chosen as a descent direction, and t_k is a proper step size determined by either exact or inexact curvilinear search conditions. Given $\rho, \varrho, \delta \in (0, 1)$, the monotone and nonmonotone Armijo rules [44] try to find the smallest integer h satisfying

$$f(R_{x_k}(t_k\eta_k)) \le C_k + \rho t_k \left\langle \mathrm{grad}\, f(x_k), \eta_k \right\rangle_{x_k}, \tag{2.2}$$

respectively, where $t_k = \gamma_k\delta^h$ and γ_k is an initial step size. Here, the reference value C_{k+1} is a convex combination of C_k and $f(x_{k+1})$ and is calculated via $C_{k+1} = (\varrho Q_k C_k + f(x_{k+1}))/Q_{k+1}$, where $C_0 = f(x_0)$, $Q_{k+1} = \varrho Q_k + 1$ and $Q_0 = 1$.

It is well known that an initial step size computed by the Barzilai–Borwein (BB) method often speeds up the convergence in Euclidean optimization. Similarly and as in [23], we can consider the following initial step sizes:

$$\gamma_k^{(1)} = \frac{\langle s_{k-1}, s_{k-1}\rangle_{x_k}}{|\langle s_{k-1}, v_{k-1}\rangle_{x_k}|} \quad \text{or} \quad \gamma_k^{(2)} = \frac{|\langle s_{k-1}, v_{k-1}\rangle_{x_k}|}{\langle v_{k-1}, v_{k-1}\rangle_{x_k}}, \tag{2.3}$$

where we can take $s_{k-1} = -t_{k-1} \cdot \mathcal{T}_{x_{k-1}\to x_k}(\mathrm{grad}\, f(x_{k-1}))$, $v_{k-1} = \mathrm{grad}\, f(x_k) + t_{k-1}^{-1} \cdot s_{k-1}$ and $\mathcal{T}_{x_{k-1}\to x_k} : \mathcal{T}_{x_{k-1}}\mathcal{M} \mapsto \mathcal{T}_{x_k}\mathcal{M}$ denotes an appropriate vector transport mapping connecting x_{k-1} and x_k; see [2, 23]. Since $\mathcal{M}$ is a submanifold of a Euclidean space, the Euclidean differences $s_{k-1} = x_k - x_{k-1}$ and $v_{k-1} = \mathrm{grad}\, f(x_k) - \mathrm{grad}\, f(x_{k-1})$ are an alternative choice if the Euclidean inner product is used in (2.3). This choice is often attractive since the vector transport is not needed; see, e.g., [39]. A variant of the nonmonotone Riemannian gradient–type method is also proposed by Iannazzo and Porcelli [23]. Global convergence to stationary points can be shown by using standard assumptions and by following the analysis in [44].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

The optimization problem (1.1) can also be solved by the proximal gradient–type method. At the kth iteration, the proximal Riemannian gradient method linearizes $f(x)$ with a proximal term to obtain the subproblem

$$\min_{x\in\mathcal{M}} \; m_k^L(x) = \langle \operatorname{grad} f(x_k), x - x_k\rangle + \frac{1}{2\tau_k}\|x - x_k\|^2, \tag{2.4}$$

where τ_k is the proximal step size and the inner products are defined in Euclidean space. It is easy to see that the solution of (2.4), denoted by x_{k+1}, is

$$x_{k+1} = \mathbf{P}_{\mathcal{M}}(x_k - \tau_k \operatorname{grad} f(x_k)) = \arg\min_{x\in\mathcal{M}} \; \frac{1}{2\tau_k}\|x - x_k + \tau_k \operatorname{grad} f(x_k)\|^2, \tag{2.5}$$

where $\mathbf{P}_{\mathcal{M}}(x) := \arg\min\{\|x - y\| : y \in \mathcal{M}\}$ is the projection operator onto $\mathcal{M}$. Notice that $\mathbf{P}_{\mathcal{M}}(x)$ exists if the manifold $\mathcal{M}$ is closed, but it may not be single-valued. Furthermore, if $\mathcal{M}$ is a submanifold of class C^2 around $\bar{x} \in \mathcal{M}$, Proposition 5 in [4] implies that $R_x(u) = \mathbf{P}_{\mathcal{M}}(x + u)$ is a retraction at x from $\mathcal{T}_x\mathcal{M}$ to $\mathcal{M}$. In this situation, the proximal gradient scheme (2.5) can be seen as a special Riemannian gradient method.

Recently, Duchi, Hazan, and Singer [14] propose the so-called Adagrad algorithm to solve online learning and stochastic optimization problems. An interesting feature of Adagrad is that it can choose different step sizes for every variable. Similarly, we can define a Riemannian version as

$$\begin{cases} G_k = G_{k-1} + \operatorname{grad} f(x_k) \odot \operatorname{grad} f(x_k), \\ x_{k+1} = \mathbf{P}_{\mathcal{M}}(x_k - \eta \operatorname{grad} f(x_k) \oslash \sqrt{G_k + \epsilon}), \end{cases} \tag{2.6}$$

where $\eta, \epsilon > 0$ and the multiplication "$\odot$" and division "$\oslash$" are performed componentwise. Motivated by the results in [14, 12], we also include and test the Riemannian Adagrad method (2.6) in our numerical experiments.

3. An adaptive quadratically regularized Newton method. Gradient-type methods often perform reasonably well but might converge slowly when the generated iterates are close to an optimal solution. Usually, fast local convergence cannot be expected if only the gradient information is used, in particular, for difficult nonquadratic or nonconvex problems. Starting from an initial point x_0, the Riemannian trust-region method [1, 2] generates the kth subproblem as follows:

$$\begin{aligned} \min_{\xi\in\mathcal{T}_{x_k}\mathcal{M}} \quad & \tilde{m}_k(\xi) := f(x_k) + \langle \operatorname{grad} f(x_k), \xi\rangle_{x_k} + \frac{1}{2}\langle \operatorname{Hess} f(x_k)[\xi], \xi\rangle_{x_k} \\ \text{s.t.} \quad & \langle \xi, \xi\rangle_{x_k} \le \Delta_k, \end{aligned} \tag{3.1}$$

where Δ_k is the trust-region radius. A common strategy is to apply the truncated preconditioned conjugate gradient method (PCG) to solve (3.1) via the linear system

$$\operatorname{grad} f(x_k) + \operatorname{Hess} f(x_k)[\xi] = 0 \tag{3.2}$$

to obtain an approximate solution ξ. The truncated PCG method terminates when either the residual becomes small enough, a negative curvature direction is detected, or the trust-region constraint is violated. Then a trial point is generated via $z_k = R_{x_k}(\xi_k)$, and the new iterate x_{k+1} is set to z_k if a certain reduction condition is satisfied. Otherwise, the iterate is not updated; i.e., it holds that $x_{k+1} := x_k$. Note

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

that (3.2) differs from the KKT condition for (3.1) since no Lagrange multiplier is involved.

We develop an adaptively regularized Riemannian Newton scheme as an alternative approach. Specifically, we use a second-order Taylor model to approximate the original objective function in the Euclidean space. Moreover, in order to control the definiteness of the model Hessian, a proximal-type penalization is added. The objective function of our subproblem is given by

$$m_k(x) := \langle \nabla f(x_k), x - x_k \rangle + \frac{1}{2} \langle H_k(x - x_k), x - x_k \rangle + \frac{\sigma_k}{2} \|x - x_k\|^2, \tag{3.3}$$

where $\nabla f(x_k)$ is the Euclidean gradient and H_k is the Euclidean Hessian of f at x_k or a suitable approximation. The regularization parameter $\sigma_k > 0$ plays a similar role as the trust-region radius Δ_k in the trust-region subproblem (3.1). A specific choice of σ_k will be discussed in subsection 3.2. We note since the model m_k is based on Euclidean gradient and Hessian information and on Euclidean structures, we primarily consider Riemannian submanifolds $\mathcal{M}$ with a corresponding, ambient Euclidean space $\mathcal{E}$ in this paper. Our overall idea now is to solve and replace the initial problem (1.1) by a sequence of subproblems of the form

$$\min_{x \in \mathcal{M}} \quad m_k(x) \tag{3.4}$$

that maintain the manifold constraints. Note that the regularized subproblem (3.4) always attains a solution whenever the manifold $\mathcal{M}$ is compact.

Similar to the classical approaches [29, 1, 2] and as in [11, 31, 38], we utilize a trust-region–like framework to monitor the acceptance of trial steps and to control the model precision by adjusting the regularization parameter σ_k. A detailed description of our method can be found in subsection 3.2. Comparing (3.4) and (2.4), our approach can also be seen as a hybrid of existing regularized trust-region algorithms [11, 31] and of the proximal Newton scheme [28] used in convex composite optimization.

In general, we do not need to solve the subproblem (3.4) exactly; we only need to find a point z_k that ensures a sufficient reduction of the model function m_k. For example, as in the classical trust-region method [29] and as in the adaptively regularized methods [11, 31], a fraction of the Cauchy decrease condition can be used to guarantee the required model decrease and to achieve global convergence. In this respect, the gradient-type methods introduced in subsection 2.1 can be ideal for solving the regularized Newton subproblems at the early stage of the algorithm when high accuracy is not needed or when a good initial guess is not available. Gradient steps can be also useful when the computational cost of evaluating the Riemannian Hessian is too expensive. When a high accuracy is required, the subproblem (3.4) can be solved more efficiently by a single or multiple Riemannian Newton steps as explained in the next subsection. Together with our specific exploitation of negative curvature information, our approach can be a good alternative to the trust-region–type methods [1, 2].

3.1. Solving the Riemannian subproblem. We use an inexact method for minimizing the model (3.4) and perform a single (or multiple) Riemannian Newton step based on the associated linear system:

$$\operatorname{grad} m_k(x_k) + \operatorname{Hess} m_k(x_k)[\xi] = 0. \tag{3.5}$$

The system (3.5) is solved approximately with a modified conjugate gradient (CG) method up to a certain accuracy. Since the model Hessian may be indefinite, we terminate the CG method when either the residual becomes small or negative curvature is detected. Then a new gradient-related direction is constructed based on the

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Algorithm 1: A Modified CG Method for Solving Subproblem (3.4)

S0 Set $T > 0$, $\theta > 1$, $\epsilon \geq 0$, $\eta_0 = 0$, $r_0 = \operatorname{grad} m_k(x_k)$, $p_0 = -r_0$, and $i = 0$.
while $i \leq n-1$ **do**
S1 Compute $\pi_i = \langle p_i, \operatorname{Hess} m_k(x_k)[p_i] \rangle_{x_k}$.
S2 **if** $\pi_i / \langle p_i, p_i \rangle_{x_k} \leq \epsilon$ **then**
if $i = 0$ **then** set $s_k = -p_0$, $d_k = 0$;
else set $s_k = \eta_i$,
if $\pi_i / \langle p_i, p_i \rangle_{x_k} \leq -\epsilon$ **then** $d_k = p_i$, set $\sigma_{est} = |\pi_i| / \langle p_i, p_i \rangle_{x_k}$;
else $d_k = 0$;
break;
S3 Set $\alpha_i = \langle r_i, r_i \rangle_{x_k} / \pi_i$, $\eta_{i+1} = \eta_i + \alpha_i p_i$, and
$r_{i+1} = r_i + \alpha_i \operatorname{Hess} m_k(x_k)[p_i]$.
S4 **if** $\|r_{i+1}\|_{x_k} \leq \min\{\|r_0\|_{x_k}^{\theta}, T\}$ **then**
choose $s_k = \eta_{i+1}$, $d_k = 0$; break;
S5 Set $\beta_{i+1} = \langle r_{i+1}, r_{i+1} \rangle_{x_k} / \langle r_i, r_i \rangle_{x_k}$ and $p_{i+1} = -r_{i+1} + \beta_{i+1} p_i$.
$i \leftarrow i + 1$.
S6 Update ξ_k according to (3.7).

conjugated directions, and a necessary curvilinear search along this direction, is utilized to reach a sufficient reduction of the objective function. The detailed procedure is presented in Algorithm 1.

We next discuss a connection between (3.5) and the classical approach (3.1)–(3.2) in the exact case $H_k = \nabla^2 f(x_k)$. In fact, the definition of the Riemannian gradient implies

$$\operatorname{grad} m_k(x_k) = \mathbf{P}_{x_k}(\nabla m_k(x_k)) = \mathbf{P}_{x_k}(\nabla f(x_k)) = \operatorname{grad} f(x_k),$$

where $\mathbf{P}_x(u) := \arg\min_{v \in \mathcal{T}_x\mathcal{M}} \|v - u\|_x$ denotes the orthogonal projection onto $\mathcal{T}_x\mathcal{M}$. Using $\nabla^2 m_k(x_k) = \nabla^2 f(x_k) + \sigma_k I$ and introducing the so-called *Weingarten map* $\mathfrak{W}_x(\cdot, v) : \mathcal{T}_x\mathcal{M} \to \mathcal{T}_x\mathcal{M}$ for some $v \in \mathcal{T}_x^{\perp}\mathcal{M}$, it holds that

$$\begin{aligned} \operatorname{Hess} m_k(x_k)[\xi] &= \mathbf{P}_{x_k}(\nabla^2 m_k(x_k)[\xi]) + \mathfrak{W}_{x_k}(\xi, \mathbf{P}_{x_k}^{\perp}(\nabla m_k(x_k))) \\ &= \operatorname{Hess} f(x_k)[\xi] + \sigma_k \xi \end{aligned} \tag{3.6}$$

for all $\xi \in \mathcal{T}_{x_k}\mathcal{M}$. The Weingarten map $\mathfrak{W}_x(\cdot, v)$ is a symmetric linear operator that is closely related to the second fundamental form of $\mathcal{M}$. The projection $\mathbf{P}_{x_k}^{\perp}$ in (3.6) is given explicitly by $\mathbf{P}_{x_k}^{\perp} = I - \mathbf{P}_{x_k}$. For a detailed derivation of the expression (3.6) and further information on the Weingarten map, we refer the reader to [3].

Although the linear systems (3.5) and (3.2) have a similar form, our approach is based on a different model formulation and uses different trial points and reduction ratios. Moreover, inspired by Steihaug's CG method [34] and by related techniques in trust-region–based optimization [18, 2], we implement a specific termination strategy whenever the CG methods encounters small or negative curvature. In particular, we utilize the detected negative curvature information to modify and improve our current search direction.

An overview of the procedure is given in Algorithm 1. Except for step S2, Algorithm 1 is a direct adaption of the CG method; see [29, Chapter 5] and [2, Chapter 7] for comparison. We generate two different output vectors s_k and d_k, where the vector d_k represents and transports the negative curvature information and s^k corresponds

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

to the "usual" output of the CG method. As specified in step S2, d^k will always be zero unless negative curvature is encountered. The new search direction ξ_k is then computed as follows:

$$\xi_k = \begin{cases} s_k + \tau_k d_k & \text{if } d_k \neq 0, \\ s_k & \text{if } d_k = 0, \end{cases} \quad \text{with} \quad \tau_k := \frac{\langle d_k, \operatorname{grad} m_k(x_k)\rangle_{x_k}}{\langle d_k, \operatorname{Hess} m_k(x_k)[d_k]\rangle_{x_k}}. \tag{3.7}$$

In section 4, we will show that ξ_k is a descent direction. Note that the rescaling factor τ_k in (3.7) can be obtained without any additional costs. The choice of τ_k is mainly motivated by our numerical experiments; see also (4.3) and [18] for a related variant.

Once the direction ξ_k is constructed, we carry out a curvilinear search along ξ_k to generate a trial point z_k, i.e.,

$$z_k = R_{x_k}(\alpha_k \xi_k). \tag{3.8}$$

The step size $\alpha_k = \alpha_0 \delta^h$ is again chosen by the (monotone) Armijo rule such that h is the smallest integer satisfying

$$m_k(R_{x_k}(\alpha_0 \delta^h \xi_k)) \leq \rho \alpha_0 \delta^h \left\langle \operatorname{grad} m_k(x_k), \xi_k \right\rangle_{x_k}, \tag{3.9}$$

where $\rho, \delta \in (0,1)$ and $\alpha_0 \in (0,1]$ are given constants.

3.2. The algorithmic framework. We now present our regularized Newton framework starting from a feasible initial point x_0 and a regularization parameter σ_0. As described in the last section, the algorithm first computes a trial point z_k to approximately solve the regularized subproblem (3.4). In order to decide whether z_k should be accepted as the next iterate and whether the regularization parameter σ_k should be updated, we calculate the ratio between the actual reduction of the objective function $f(x)$ and the predicted reduction:

$$\rho_k = \frac{f(z_k) - f(x_k)}{m_k(z_k)}. \tag{3.10}$$

If $\rho_k \geq \eta_1 > 0$, then the iteration is successful, and we set $x_{k+1} = z_k$; otherwise, the iteration is not successful, and we set $x_{k+1} = x_k$; i.e., we have

$$x_{k+1} = \begin{cases} z_k & \text{if } \rho_k \geq \eta_1, \\ x_k & \text{otherwise.} \end{cases} \tag{3.11}$$

The regularization parameter σ_{k+1} is updated as follows:

$$\sigma_{k+1} \in \begin{cases} (0, \gamma_0 \sigma_k] & \text{if } \rho_k \geq \eta_2, \\ [\gamma_0 \sigma_k, \gamma_1 \sigma_k] & \text{if } \eta_1 \leq \rho_k < \eta_2, \\ [\gamma_1 \sigma_k, \gamma_2 \sigma_k] & \text{otherwise,} \end{cases} \tag{3.12}$$

where $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_0 < 1 < \gamma_1 \leq \gamma_2$. These parameters determine how aggressively the regularization parameter is adjusted when an iteration is successful or unsuccessful. The complete regularized Newton method to solve (1.1) is summarized in Algorithm 2.

At this point, let us notice that the regularized subproblem (3.4) might be unbounded in general. However, this is not really problematic in our algorithmic framework since the modified CG Algorithm 1 will detect a small or negative curvature direction and terminate after a finite number of steps in such a case. Hence, step S1—the calculation of the trial point z_k—in Algorithm 2 is always well-defined even if the corresponding subproblem is unbounded.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Algorithm 2: An Adaptive Regularized Newton Method

S0 Choose a feasible initial point $x_0 \in \mathcal{M}$ and an initial regularization parameter $\sigma_0 > 0$. Choose $0 < \eta_1 \leq \eta_2 < 1$, $0 < \gamma_0 < 1 < \gamma_1 \leq \gamma_2$. Set $k := 0$.

while *stopping conditions not met* **do**

S1 Compute a new trial point z_k according to (3.8) and (3.9).

S2 Compute the ratio ρ_k via (3.10).

S3 Update x_{k+1} from the trial point z_k based on (3.11).

S4 Update σ_k according to (3.12).

$k \leftarrow k + 1$.

4. Convergence analysis. We now analyze the convergence of Algorithm 2 based on the model (3.4). Let us note that the analysis can be similarly extended to the algorithm using cubically regularized subproblems as well. In the following, we summarize and present our main assumptions.

Assumption 1. Let $\{x_k\}$ be generated by Algorithm 2. We assume the following:

(A.1) The gradient ∇f is Lipschitz continuous on the convex hull of the manifold $\mathcal{M}$, denoted by $\operatorname{conv}(\mathcal{M})$; i.e., there exists $L_f > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\| \quad \forall\, x, y \in \operatorname{conv}(\mathcal{M}).$$

(A.2) There exists $\kappa_g > 0$ such that $\|\nabla f(x_k)\| \leq \kappa_g$ for all $k \in \mathbb{N}$.

(A.3) There exists $\kappa_H > 0$ such that $\|H_k\| \leq \kappa_H$ for all $k \in \mathbb{N}$.

(A.4) The Euclidean and the Riemannian Hessian are bounded; i.e., there exist κ_F and $\kappa_R \geq 1$ such that

$$\|\nabla^2 f(x_k)\| \leq \kappa_F \quad \text{and} \quad \|\operatorname{Hess} f(x_k)\|_{x_k} \leq \kappa_R \quad \forall\, k \in \mathbb{N}.$$

(A.5) Let $\varpi_{x_k}^m$, $\varpi_{x_k}^M$ be given as in (2.1). Then suppose there exists $\underline{\varpi} > 0$, $\overline{\varpi} \geq 1$ such that $\underline{\varpi} \leq \varpi_{x_k}^m$ and $\varpi_{x_k}^M \leq \overline{\varpi}$ for all $k \in \mathbb{N}$.

Remark 2. Suppose that the level set $\mathcal{L} := \{x \in \mathcal{M} : f(x) \leq f(x_0)\}$ is compact. Then by construction of Algorithm 2, we have $f(x_{k+1}) = f(x_k) + \rho_k m_k(z_k) \leq f(x_k)$ if iteration k is successful. Due to (3.11), it follows that $x_k \in \mathcal{L}$ for all k, and the sequence $\{x_k\}$ must be bounded. Hence, in this case, the assumptions (A.2) and (A.4) hold automatically. Furthermore, since the parameters $\varpi_{x_k}^m$, $\varpi_{x_k}^M$, $k \in \mathbb{N}$, depend continuously on x_k, assumption (A.5) is also satisfied.

Remark 3. The bounds in Assumption 1 can also be used to derive a bound for $\operatorname{Hess} m_k(x_k)$. In fact, under the conditions (A.3)–(A.4) and by (3.6), we have

$$\begin{aligned}\langle \xi, \operatorname{Hess} m_k(x_k)[\xi] \rangle_{x_k} &= \big\langle \xi, \operatorname{Hess} f(x_k)[\xi] + \mathbf{P}_{x_k}((H_k - \nabla^2 f(x_k))[\xi]) \big\rangle_{x_k} + \sigma_k \|\xi\|_{x_k}^2 \\ &\leq (\kappa_R + (\varpi_{x_k}^M)^{\frac{1}{2}} (\varpi_{x_k}^m)^{-\frac{1}{2}} (\kappa_H + \kappa_F) + \sigma_k) \|\xi\|_{x_k}^2,\end{aligned}$$

where we used the linearity and nonexpansiveness of the operator $\mathbf{P}_{x_k}$. In the following, we set $\kappa_{x_k}^M := \kappa_R + (\varpi_{x_k}^M)^{\frac{1}{2}} (\varpi_{x_k}^m)^{-\frac{1}{2}} (\kappa_H + \kappa_F)$.

4.1. Analysis of the inner subproblem. At first, we briefly discuss several useful properties of the modified CG method. In the following, we also consider the m_k-based model function

$$\breve{m}_k(\xi) := f(x_k) + \langle \operatorname{grad} m_k(x_k), \xi \rangle_{x_k} + \frac{1}{2} \langle \operatorname{Hess} m_k(x_k)[\xi], \xi \rangle_{x_k}, \quad \xi \in \mathcal{T}_{x_k}\mathcal{M}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

As we have shown in subsection 3.1 and (3.6), the model functions $\tilde{m}_k$ and $\breve{m}_k$ coincide if the exact Hessian $H_k = \nabla^2 f(x_k)$ is used and we set $\sigma_k = 0$.

LEMMA 4. *Let the sequences* $\{p_i\}_{i=0}^{\ell}$, $\{r_i\}_{i=0}^{\ell}$, $\{\eta_i\}_{i=0}^{\ell}$ *and the direction* ξ_k *be generated by Algorithm* 1. *Then we have the following:*

(i) *For all* $j = 1, \ldots, \ell$, *it holds that* $p_j \in \mathcal{T}_{x_k}\mathcal{M}$,

$$\langle p_j, \operatorname{Hess} m_k(x_k)[p_i]\rangle_{x_k} = 0 \quad \text{and} \quad \langle r_j, r_i\rangle_{x_k} = 0 \quad \forall\, i = 0, \ldots, j-1. \tag{4.1}$$

(ii) *The sequence* $\{\breve{m}_k(\eta_i)\}$ *is strictly decreasing, and it holds that* $\breve{m}_k(\xi_k) < \breve{m}_k(\eta_\ell)$.

(iii) *The sequence* $\{\|\eta_i\|_{x_k}\}$ *is strictly increasing, and it holds that* $\|\xi_k\|_{x_k} \geq \|\eta_\ell\|_{x_k}$.

Proof. Except for step S2, Algorithm 1 coincides with the standard CG method applied to the quadratic problem $\min_\xi \breve{m}_k(\xi)$. Since $\operatorname{Hess} m_k(x_k)$ is a linear operator from $\mathcal{T}_{x_k}\mathcal{M}$ to $\mathcal{T}_{x_k}\mathcal{M}$, all iterates generated by Algorithm 1 will stay in the tangent space $\mathcal{T}_{x_k}\mathcal{M}$. Furthermore, since the Riemannian Hessian is symmetric with respect to the metric $\langle\cdot,\cdot\rangle_{x_k}$ [2, Proposition 5.5.3], parts (i) and (ii) essentially follow from the properties of the CG method in Euclidean space. We refer the reader to [29, section 5.1] for further details. If $d_k \neq 0$, the estimate $\breve{m}_k(\xi_k) \leq \breve{m}_k(\eta_\ell)$ follows from (4.1) and $\pi_\ell < 0$. The first claim in (iii) is proven in [29, Theorem 7.3]. To verify $\|\xi_k\|_{x_k} \geq \|\eta_\ell\|_{x_k}$, we first show

$$\|r_i\|_{x_k}^2 = -\langle \operatorname{grad} m_k(x_k), p_i\rangle_{x_k} \quad \forall\, i = 0, \ldots, \ell-1 \tag{4.2}$$

by induction. For $i = 0$, (4.2) is obviously satisfied by definition of r_0 and p_0. Now let us suppose that (4.2) holds for $i = \ell - 1$. Then by (4.1), we have

$$-\langle \operatorname{grad} m_k(x_k), p_\ell\rangle_{x_k} = \langle r_0, r_\ell - \beta_\ell p_{\ell-1}\rangle_{x_k} = -\frac{\|r_\ell\|_{x_k}^2}{\|r_{\ell-1}\|_{x_k}^2}\langle r_0, p_{\ell-1}\rangle_{x_k} = \|r_\ell\|_{x_k}^2.$$

Thus, if $d_k \neq 0$, this implies

$$\xi_k = \eta_\ell + \tau_k d_k = \sum_{i=0}^{\ell-1}\alpha_i p_i - \frac{\|r_\ell\|_{x_k}^2}{\pi_\ell} p_\ell = \sum_{i=0}^{\ell}|\alpha_i| p_i. \tag{4.3}$$

Consequently, since ξ_k and η_ℓ coincide in the case $d_k = 0$, the estimate $\|\xi_k\|_{x_k} \geq \|\eta_\ell\|_{x_k}$ again follows from [29, Theorem 7.3] (and from the special structure of ξ_k). ∎

We now prove that the direction ξ_k is a descent direction.

LEMMA 5. *Let* $\{\alpha_i\}, \{\pi_i\}, \{p_i\}$, *and* $\{\eta_i\}$ *be generated by Algorithm* 1, *and suppose that the conditions* (A.3)–(A.4) *are satisfied. Then the direction* ξ_k *—given in* (3.7)*—is a descent direction, and it holds that*

$$\frac{\langle \operatorname{grad} m_k(x_k), \xi_k\rangle_{x_k}}{\|\operatorname{grad} m_k(x_k)\|_{x_k}\|\xi_k\|_{x_k}} \leq -\min\left\{\frac{\epsilon}{2}, 1\right\}\frac{1}{n(\kappa_{x_k}^M + 1)} =: -\lambda_k. \tag{4.4}$$

Proof. We first analyze the case where Algorithm 2 detects a small or negative curvature and terminates in step S2. In this situation, we have

$$\xi_k = s_k + \tau_k d_k = \begin{cases} -\operatorname{grad} m_k(x_k) & \text{if } \ell = 0 \text{ and } \pi_\ell \leq \epsilon\|p_\ell\|_{x_k}^2, \\ \eta_\ell & \text{if } \ell > 0 \text{ and } |\pi_\ell| \leq \epsilon\|p_\ell\|_{x_k}^2, \\ \eta_\ell + \tau_k p_\ell & \text{if } \ell > 0 \text{ and } \pi_\ell < -\epsilon\|p_\ell\|_{x_k}^2, \end{cases}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

with $\tau_k = \pi_\ell^{-1} \langle \operatorname{grad} m_k(x_k), p_\ell \rangle_{x_k}$. We note that condition (4.4) is obviously satisfied with $\lambda_k := 1$ in the case $\ell = 0$. Next, let us consider the case $\ell > 0$ and $\pi_\ell < -\epsilon \|p_\ell\|_{x_k}^2$. Due to (4.2), we have

$$\langle \operatorname{grad} m_k(x_k), \eta_\ell \rangle_{x_k} = \sum_{i=0}^{\ell-1} \alpha_i \langle \operatorname{grad} m_k(x_k), p_i \rangle_{x_k} = -\sum_{i=0}^{\ell-1} \frac{\langle \operatorname{grad} m_k(x_k), p_i \rangle_{x_k}^2}{\pi_i}$$

and thus

$$\langle \operatorname{grad} m_k(x_k), \xi_k \rangle_{x_k} = -\sum_{i=0}^{\ell} \frac{\langle \operatorname{grad} m_k(x_k), p_i \rangle_{x_k}^2}{|\pi_i|} \leq -\frac{\|p_0\|_{x_k}^4}{\pi_0} \leq -\frac{\|\operatorname{grad} m_k(x_k)\|_{x_k}^2}{\kappa_{x_k}^M + \sigma_k},$$

where we used the conditions (A.3)–(A.4), Remark 3, and $\pi_i > 0$, $i = 0, \ldots, \ell - 1$. By construction of the algorithm, it holds that $|\pi_i| = \pi_i > \epsilon \|p_i\|_{x_k}^2$ for all $i = 0, \ldots, \ell - 1$ and $|\pi_\ell| = -\pi_\ell > \epsilon \|p_\ell\|_{x_k}^2$. Hence, we obtain

$$\begin{aligned} \|\xi_k\|_{x_k} &\leq \sum_{j=0}^{\ell} \frac{|\langle \operatorname{grad} m_k(x_k), p_i \rangle_{x_k}|}{|\pi_i|} \cdot \|p_i\|_{x_k} \\ &\leq (\ell + 1) \|\operatorname{grad} m_k(x_k)\|_{x_k} \cdot \max_{i \in \{0, \ldots, \ell\}} \frac{\|p_i\|_{x_k}^2}{|\pi_i|} \leq \frac{n}{\epsilon} \cdot \|\operatorname{grad} m_k(x_k)\|_{x_k}. \end{aligned}$$

Moreover, if $\sigma_k \geq \kappa_{x_k}^M + 2$, then we have $|\pi_i| \|p_i\|_{x_k}^{-2} \geq \sigma_k - \kappa_{x_k}^M > 0$, and the last estimate becomes $\|\xi_k\|_{x_k} \leq n(\sigma_k - \kappa_{x_k}^M)^{-1} \|\operatorname{grad} m_k(x_k)\|_{x_k}$. Combining these results, we now get

$$\frac{\langle \operatorname{grad} m_k(x_k), \xi_k \rangle_{x_k}}{\|\operatorname{grad} m_k(x_k)\|_{x_k} \|\xi_k\|_{x_k}} \leq -\min\left\{\frac{\epsilon}{2}, 1\right\} \frac{1}{n(\kappa_{x_k}^M + 1)}.$$

Due to the special structure of ξ_k (see again (4.3)), the same estimates can also be used and derived in the remaining cases. This finishes the proof. ∎

In the next lemma, we prove that the descent property of ξ_k can be carried over to the Euclidean model m_k using the smooth retraction R and that a sufficient reduction of the objective function m_k in the sense of (3.9) can be ensured.

LEMMA 6. *Suppose that the assumptions* (A.2)–(A.4) *are satisfied. Let* $\rho \in (0, 1)$ *be arbitrary, and set* $z_k(t) := R_{x_k}(t\xi_k)$. *Then we have*

$$m_k(z_k(t)) \leq \rho t \langle \operatorname{grad} m_k(x_k), \xi_k \rangle_{x_k} \quad \forall\ t \in [0, \zeta_k], \tag{4.5}$$

where

$$\zeta_k := \min\left\{(\varpi_{x_k}^M)^{-1}, 1\right\} \min\left\{ \frac{\chi}{\|\xi_k\|_{x_k}}, \frac{2(1-\rho)\lambda_k}{(\kappa_2 \kappa_g + \kappa_1^2(\kappa_H + \sigma_k))} \frac{\|\operatorname{grad} f(x_k)\|_{x_k}}{\|\xi_k\|_{x_k}} \right\} \tag{4.6}$$

and κ_1, κ_2, χ *are constants that do not depend on* x_k.

Proof. Let us set $\phi(t) := m_k(R_{x_k}(t\xi_k))$. Then, since $\mathcal{M}$ is an embedded submanifold and using the properties of the retraction R_{x_k}, it follows that

$$\begin{aligned} m_k(R_{x_k}(t\xi_k)) &= \phi(0) + t\phi'(0) + \int_0^t \phi'(s) - \phi'(0)\, \mathrm{ds} \\ &= t \langle \nabla f(x_k), \xi_k \rangle + \int_0^t \langle \nabla f(x_k), (DR_{x_k}(s\xi_k) - \mathrm{id})[\xi_k] \rangle \\ &\qquad + \langle (R_{x_k}(s\xi_k) - x_k), (H_k + \sigma_k I)[DR_{x_k}(s\xi_k)\xi_k] \rangle\, \mathrm{ds}, \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\mathrm{id} \equiv \mathrm{id}_{\mathcal{T}_x\mathcal{M}}$ denotes the identity mapping on $\mathcal{T}_x\mathcal{M}$. As in [7, section B], we define the compact set $K_\chi := \{\xi \in \mathcal{TM} : \|\xi\| \leq \chi\}$. The smoothness of R now implies

$$\|R_{x_k}(\xi) - x_k\| \leq \int_0^1 \|DR_{x_k}(s\xi)[\xi]\| \,\mathrm{ds} \leq \max_{y \in K_\chi} \|DR(y)\|\|\xi\| \tag{4.7}$$

and

$$\|DR_{x_k}(\xi) - \mathrm{id}\| \leq \int_0^1 \|D^2R_{x_k}(s\xi)[\xi]\| \,\mathrm{ds} \leq \max_{y \in K_\chi} \|D^2R(y)\|\|\xi\| \tag{4.8}$$

for all $\xi \in K_\chi$. Setting $\kappa_1 := \max_{y \in K_\chi} \|DR(y)\|$ and $\kappa_2 := \max_{y \in K_\chi} \|D^2R(y)\|$ and using the assumptions (A.2)–(A.3), this yields

$$\begin{aligned} m_k(R_{x_k}(t\xi_k)) &\leq t\,\langle \nabla f(x_k), \xi_k\rangle + \int_0^t (\kappa_2\kappa_g + \kappa_1^2(\kappa_H + \sigma_k))s\|\xi_k\|^2 \,\mathrm{ds} \\ &= t\,\langle \mathrm{grad}\, f(x_k), \xi_k\rangle_{x_k} + \frac{1}{2}(\kappa_2\kappa_g + \kappa_1^2(\kappa_H + \sigma_k))t^2\|\xi_k\|^2 \end{aligned}$$

if $t\|\xi_k\| \leq \chi$. Thus, by Lemma 5 and setting $\kappa := \kappa_2\kappa_g + \kappa_1^2(\kappa_H + \sigma_k)$, we obtain

$$\begin{aligned} &m_k(R_{x_k}(t\xi_k)) - \rho t\,\langle \mathrm{grad}\, m(x_k), \xi_k\rangle_{x_k} \\ &\qquad \leq -(1-\rho)\lambda_k t\|\mathrm{grad}\, f(x_k)\|_{x_k}\|\xi_k\|_{x_k} + \frac{1}{2}\kappa\varpi_{x_k}^M t^2\|\xi_k\|_{x_k}^2 \\ &\qquad \leq \left[\frac{1}{2}\kappa\varpi_{x_k}^M t - (1-\rho)\lambda_k \frac{\|\mathrm{grad}\, f(x_k)\|_{x_k}}{\|\xi_k\|_{x_k}}\right] \cdot t\|\xi_k\|_{x_k}^2 \end{aligned}$$

if $t\|\xi_k\| \leq \chi$. Finally, using the last estimate, (2.1), and $(\varpi_{x_k}^M)^{\frac{1}{2}} \leq \max\{\varpi_{x_k}^M, 1\}$, this establishes (4.5) and (4.6). ∎

4.2. Global convergence. In this section, based on the techniques used in [11], we present global convergence properties of the adaptive regularized Newton method. We first investigate the relationship between the reduction ratio ρ_k defined in (3.10), the regularization parameter σ_k, and the gradient norm $\|\mathrm{grad}\, f(x_k)\|_{x_k}$. Under the assumption $\|\mathrm{grad}\, f(x_k)\|_{x_k} \geq \tau > 0$, we then derive an upper bound for σ_k and show that the iterations will be successful (i.e., $\rho_k \geq \eta_1$) whenever σ_k exceeds this bound. In Theorem 11 we combine our observations and establish convergence of our method.

The next lemma shows that the distance between z_k and x_k is bounded by some value related to the regularization parameter σ_k.

LEMMA 7. *Suppose that the assumptions* (A.2)–(A.3) *hold and that* z_k *satisfies the Armijo condition* (3.9). *Then it holds that*

$$\|z_k - x_k\| \leq \frac{2\kappa_g}{\sigma_k - \kappa_H}$$

whenever $\sigma_k > \kappa_H$.

Proof. By Lemma 5 we have $m_k(z_k) \leq 0$. Thus, it follows that

$$\langle \nabla f(x_k), z_k - x_k\rangle + \frac{1}{2}\langle z_k - x_k, H_k[z_k - x_k]\rangle + \frac{\sigma_k}{2}\|z_k - x_k\|^2 \leq 0.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

If $\sigma_k \geq \kappa_H$, then the term $\|z_k - x_k\|$ can be bounded by

$$-\|\nabla f(x_k)\|\|z_k - x_k\| - \frac{1}{2}\kappa_H\|z_k - x_k\|^2 + \frac{1}{2}\sigma_k\|z_k - x_k\|^2 \leq 0,$$

and hence

$$\|z_k - x_k\| \leq \frac{2\|\nabla f(x_k)\|}{\sigma_k - \kappa_H} \leq \frac{2\kappa_g}{\sigma_k - \kappa_H}. \qquad \square$$

When the regularization parameter is sufficiently large, our model defines a good approximation of the initial problem (1.1). In this case, a successful iteration and sufficient reduction of the objective function can be ensured.

LEMMA 8. *Suppose that the conditions* (A.1)–(A.4) *hold and that* z_k *satisfies the Armijo condition* (3.9). *Furthermore, let us assume* $g_k := \|\operatorname{grad} f(x_k)\|_{x_k} \neq 0$ *and*

$$\sigma_k \geq \max\left\{\kappa_{x_k}^M, \kappa_H + \vartheta_k \max\left\{\frac{1}{\sqrt{\chi}}, \frac{\sqrt{A_2^k}}{\sqrt{g_k}}, \frac{A_3^k\vartheta_k}{g_k}\right\}\right\}, \quad \vartheta_k := \sqrt{\frac{A_1^k \max\{\varpi_{x_k}^M, 1\}}{(1-\eta_2)g_k}},$$

where $\kappa := \kappa_2\kappa_g + 2\kappa_1^2\kappa_H$, $A_1^k := 2\kappa_g^2\alpha_0(\rho\lambda_k\delta)^{-1}(L_f + \kappa_H)$, $A_2^k := ((1-\rho)\lambda_k)^{-1}\kappa$, *and* $A_3^k := ((1-\rho)\lambda_k)^{-1}\kappa_1^2$. *Then iteration* k *is very successful; i.e., it holds that* $\rho_k \geq \eta_2$ *and* $\sigma_{k+1} \leq \gamma_0\sigma_k$.

Proof. Using the Lipschitz continuity of ∇f and (A.3), it follows that

$$\begin{aligned} f(z_k) - f(x_k) - m_k(z_k) &= \langle \nabla f(x_k + \tau(z_k - x_k)) - \nabla f(x_k), z_k - x_k\rangle \\ &\quad - \frac{1}{2}\langle H_k[z_k - x_k], z_k - x_k\rangle - \frac{\sigma_k}{2}\|z_k - x_k\|^2 \\ &\leq \frac{1}{2}(L_f + \kappa_H)\|z_k - x_k\|^2 \end{aligned}$$

for some $\tau \in (0,1)$. Applying Lemmas 6 and 7 and the Armijo condition (3.9), we now obtain

$$\begin{aligned} 1 - \rho_k &= \frac{f(z_k) - f(x_k) - m_k(z_k)}{-m_k(z_k)} \\ &\leq \frac{(L_f + \kappa_H)\|z_k - x_k\|^2}{2\rho\lambda_k\alpha_k\|\operatorname{grad} f(x_k)\|_{x_k}\|\xi_k\|_{x_k}} \\ &\leq \frac{2(L_f + \kappa_H)\kappa_g^2}{\rho\lambda_k\alpha_0^{-1}\delta} \cdot \frac{\max\{\varpi_{x_k}^M, 1\}}{(\sigma_k - \kappa_H)^2 g_k} \cdot \max\left\{\frac{1}{\chi}, \frac{\kappa_2\kappa_g + \kappa_1^2(\sigma_k + \kappa_H)}{2(1-\rho)\lambda_k g_k}\right\} \\ &\leq \frac{A_1^k \max\{\varpi_{x_k}^M, 1\}}{(\sigma_k - \kappa_H)^2 g_k} \max\left\{\frac{1}{\chi}, \frac{\kappa}{(1-\rho)\lambda_k g_k}, \frac{\kappa_1^2(\sigma_k - \kappa_H)}{(1-\rho)\lambda_k g_k}\right\} \\ &= \frac{A_1^k \max\{\varpi_{x_k}^M, 1\}}{(\sigma_k - \kappa_H)^2 g_k} \max\left\{\frac{1}{\chi}, \frac{A_2^k}{g_k}, \frac{A_3^k}{g_k}(\sigma_k - \kappa_H)\right\} \\ &\leq 1 - \eta_2. \end{aligned}$$

The above inequality shows $\rho_k \geq \eta_2$. Finally, step S4 of Algorithm 2 implies $\sigma_{k+1} \leq \gamma_0\sigma_k$, as desired. $\square$

We next prove that the regularization parameters can be bounded.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

LEMMA 9. *Suppose that the assumptions* (A.1)–(A.5) *are satisfied and there exists* $\tau > 0$ *such that* $\|\operatorname{grad} f(x_k)\|_{x_k} \geq \tau$ *for all* $k \in \mathbb{N}$. *Then, the sequence* $\{\sigma_k\}$ *is bounded, i.e., there exists* $L_\tau \geq 0$ *such that*

$$\sigma_k \leq L_\tau, \quad \forall\, k \in \mathbb{N}. \tag{4.9}$$

Proof. At first, using the bounds in (A.5), it holds that $\kappa_{x_k}^M \leq \kappa_R + (\overline{\varpi})^{\frac{1}{2}} (\underline{\varpi})^{-\frac{1}{2}}(\kappa_H + \kappa_F) =: \bar{\kappa}_M$. Hence, it follows that $\lambda_k \geq \min\left\{\frac{\epsilon}{2}, 1\right\}(n(\bar{\kappa}_M + 1))^{-1} =: \bar{\lambda}$ and similarly

$$A_1^k \leq (2\kappa_g^2 \alpha_0 (L_f + \kappa_H))(\rho\delta\bar{\lambda})^{-1} =: A_1, \quad A_2^k \leq \kappa((1-\rho)\bar{\lambda})^{-1} =: A_2$$

and $A_3^k \leq \kappa_1^2((1-\rho)\bar{\lambda})^{-1} =: A_3$. We now define

$$\kappa_\tau := \max\left\{\bar{\kappa}_M, \kappa_H + \vartheta_\tau \max\left\{\frac{1}{\sqrt{\chi}}, \frac{\sqrt{A_2}}{\sqrt{\tau}}, \frac{A_3\vartheta_\tau}{\tau}\right\}\right\}, \quad \vartheta_\tau := \sqrt{\frac{A_1\overline{\varpi}}{(1-\eta_2)\tau}}.$$

Let us assume that the bound $\sigma_k \geq \kappa_\tau$ holds for some $k \geq 0$. Then Lemma 8 implies that iteration k is very successful with $\sigma_{k+1} \leq \sigma_k$. Consequently, when $\sigma_0 \leq \gamma_2\kappa_\tau$, we have $\sigma_k \leq \gamma_2\kappa_\tau$, $k \geq 0$, where the factor γ_2 is introduced to cover the case that σ_k is less than κ_τ and iteration k is not very successful. Setting $L_\tau := \max\{\sigma_0, \gamma_2\kappa_\tau\}$, we obtain (4.9). ∎

Based on the results in [11] and [2, section 7] and similar to [31], we now show global convergence of our adaptive regularized Newton method. We first analyze the behavior of Algorithm 2 under the assumption that only finitely many successful iterations are performed.

LEMMA 10. *Suppose that the assumptions* (A.1)–(A.5) *are satisfied and there are only finitely many successful iterations. Then it holds* $x_k = x_*$ *for all sufficiently large* k *and* $\operatorname{grad} f(x_*) = 0$.

Proof. Let the last successful iteration be indexed by ℓ; then, due to the construction of Algorithm 2, it holds that $x_{\ell+1} = x_k = x_*$ for all $k \geq \ell+1$. Since all iterations $k \geq \ell+1$ are unsuccessful, the regularization parameter σ_k tends to infinity as $k \to \infty$. If $\|\operatorname{grad} f(x_{\ell+1})\|_{x_{\ell+1}} > 0$, then we have $\|\operatorname{grad} f(x_k)\|_{x_k} = \|\operatorname{grad} f(x_{\ell+1})\|_{x_{\ell+1}} > 0$ for all $k \geq \ell+1$, and Lemma 9 implies that σ_k is bounded above, $k \geq \ell+1$. This contradiction completes the proof. ∎

The following theorem generalizes [11, Theorem 2.5] and represents our main convergence result in this section.

THEOREM 11. *Suppose that the assumptions* (A.1)–(A.5) *hold, and let* $\{f(x_k)\}$ *be bounded from below. Then either*

$$\operatorname{grad} f(x_\ell) = 0 \;\; \textit{for some} \;\; \ell \geq 0 \quad \textit{or} \quad \liminf_{k\to\infty} \|\operatorname{grad} f(x_k)\|_{x_k} = 0.$$

Proof. Due to Lemma 10, we only have to consider the case when infinitely many successful iterations occur. Let us assume that there exists $\tau > 0$ such that

$$\|\operatorname{grad} f(x_k)\|_{x_k} \geq \tau \quad \forall\, k \geq 0, \tag{4.10}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

and let $k \in \mathcal{S}$ with $\mathcal{S} := \{k \in \mathbb{N} : \text{iteration } k \text{ is successful or very successful}\}$ be given. As in the proof of Lemma 9, there exists $\bar{\lambda}$ such that $\lambda_k \geq \bar{\lambda}$ for all $k \in \mathbb{N}$. Now Lemmas 6 and 9 imply that

$$\begin{aligned} f(x_k) - f(z_k) &\geq \eta_1 \cdot (-m_k(z_k)) \geq \eta_1 \rho \bar{\lambda} \alpha_0^{-1} \delta \zeta_k \cdot \|\operatorname{grad} f(x_k)\|_{x_k} \|\xi_k\|_{x_k} \\ &\geq \eta_1 \rho \bar{\lambda} \delta (\alpha_0 \overline{\omega})^{-1} \tau \cdot \min\left\{ \chi, \frac{2(1-\rho)\bar{\lambda}\tau}{\kappa + \kappa_1^2 (L_\tau - \kappa_H)} \right\} =: \delta_\tau. \end{aligned}$$

Summing up over all iterates yields

$$f(x_0) - f(x_{k+1}) = \sum_{j=0, j \in \mathcal{S}}^{k} f(x_j) - f(x_{j+1}) \geq |\mathcal{S} \cap \{1, \ldots, k\}| \cdot \delta_\tau. \tag{4.11}$$

Since $\mathcal{S}$ is not finite, we have $|\mathcal{S} \cap \{1, \ldots, k\}| \to \infty$ as $k \to \infty$. Consequently, inequality (4.11) implies $\lim_{k\to\infty} f(x_0) - f(x_{k+1}) = \infty$, which is a contradiction to the lower boundedness of $\{f(x_k)\}$. Hence, assumption (4.10) must be false, and $\{\|\operatorname{grad} f(x_k)\|_{x_k}\}$ has a subsequence that converges to zero. ∎

Remark 12. Let us note that it is possible to obtain a slightly stronger result and establish convergence of the full sequence $\|\operatorname{grad} f(x_k)\|_{x_k} \to 0$ as $k \to \infty$. However, this requires additional assumptions on the retraction and on the Lipschitz continuity of the Riemannian gradient. We refer to [31, Corollary 4.2.1] for a related discussion and result.

4.3. Local convergence. In this part, we analyze the local convergence properties of Algorithm 2. Because our inner solver is a regularized Newton method, the local superlinear convergence can be established using similar techniques as in the standard trust-region method [29]. Following [2, Proposition 7.4.5], we first present an assumption on the boundedness of the second-order covariant derivatives $\frac{D}{dt}\frac{d}{dt}R$ of the retraction R.

Assumption 13. Suppose that there exists $\beta_R, \delta_R > 0$ such that

$$\left\| \frac{D}{dt} \frac{d}{dt} R_x(t\xi) \right\|_x \leq \beta_R$$

for all $x \in \mathcal{M}$, all $\xi \in \mathcal{T}_x\mathcal{M}$ with $\|\xi\|_x = 1$, and all $t < \delta_R$.

We refer to [2, Chapter 5] for a detailed discussion of covariant derivatives. Let us note that Assumption 13 is satisfied whenever the manifold $\mathcal{M}$ is compact; see, e.g., [2, Corollary 7.4.6]. We now present our main assumptions that are necessary to prove fast local convergence of Algorithm 2. Let us emphasize that our assumptions are similar to the ones used in other Riemannian optimization frameworks.

Assumption 14. Let $\{x_k\}$ be generated by Algorithm 2. We assume the following:

(B.1) The sequence $\{x_k\}$ converges to x_*.

(B.2) The Euclidean Hessian $\nabla^2 f$ is continuous on $\operatorname{conv}(\mathcal{M})$.

(B.3) The Riemannian Hessian $\operatorname{Hess} f$ is positive definite at x_*, and the constant ϵ in Algorithm 1 is set to zero.

(B.4) The matrices H_k, $k \in \mathbb{N}$, satisfy the following Dennis–Moré condition:

$$\frac{\|(H_k - \nabla^2 f(x_k))[z_k - x_k]\|}{\|z_k - x_k\|} \to 0 \quad \text{whenever} \quad \|\operatorname{grad} f(x_k)\|_{x_k} \to 0.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(B.5) H_k is a good approximation of the Euclidean Hessian $\nabla^2 f$; i.e., it holds that

$$\|H_k - \nabla^2 f(x_k)\| \to 0 \quad \text{whenever} \quad \|\operatorname{grad} f(x_k)\|_{x_k} \to 0.$$

In the following lemma and inspired by [11, Theorem 4.3] and [31, Theorem 4.2.2], we show that the iterations generated by Algorithm 2 are eventually very successful. Due to (3.12), this also implies that the sequence of regularization parameters $\{\sigma_k\}$ converges to zero as $k \to \infty$.

LEMMA 15. *Let the conditions* (A.3) *and* (B.1)–(B.4) *be satisfied. Then all iterations are eventually very successful.*

Proof. First, Theorem 11 implies that x_* is stationary point of problem (1.1); i.e., we have $\operatorname{grad} f(x_k) \to \operatorname{grad} f(x_*) = 0$ as $k \to \infty$. Moreover, since $\{x_k\}$ converges to x_*, the assumptions (A.2) and (A.4)–(A.5) are satisfied. We next use a connection between ξ_k and $\operatorname{grad} f(x_k)$ that was established in the proof of Lemma 5; it holds that

$$\|\xi_k\|_{x_k} \leq \min\{\epsilon^{-1}, 1\} n \cdot \|\operatorname{grad} f(x_k)\|_{x_k} \to 0, \quad k \to \infty. \tag{4.12}$$

Hence, we have $\|\xi_k\| \leq \chi$ for all k sufficiently large, and thus, from (4.7), it follows that

$$\|z_k - x_k\| \leq \kappa_1 \alpha_k \|\xi_k\| \leq \min\{\epsilon^{-1}, 1\} n \kappa_1 \sqrt{\overline{\varpi}} \cdot \|\operatorname{grad} f(x_k)\|_{x_k}. \tag{4.13}$$

Similar to [7, section B] and by combining (4.7)–(4.8), we obtain

$$\|z_k - x_k - \alpha_k \xi_k\| = \|R_{x_k}(\alpha_k \xi_k) - x_k - \alpha_k \xi_k\| \leq \frac{\overline{\varpi}\kappa_2}{2} \alpha_k^2 \|\xi_k\|_{x_k}^2 \tag{4.14}$$

for all k sufficiently large. Using the continuity of the Riemannian Hessian and (B.3), there exists $\nu > 0$ such that $\langle \xi, \operatorname{Hess} f(x_k)[\xi] \rangle_{x_k} \geq \nu \|\xi\|_{x_k}^2$ for all $\xi \in \mathcal{T}_{x_k}\mathcal{M}$ and $k \in \mathbb{N}$ sufficiently large. Setting $m_k^F(x) := m_k(x) - \frac{\sigma_k}{2}\|x - x_k\|^2$, this implies that

$$\langle \xi_k, \operatorname{Hess} m_k(x_k)[\xi_k] \rangle_{x_k} \geq (\nu + \sigma_k)\|\xi_k\|_{x_k}^2 - \left| \left\langle \xi_k, (\operatorname{Hess} m_k^F(x_k) - \operatorname{Hess} f(x_k))[\xi_k] \right\rangle_{x_k} \right|.$$

Due to (3.6), we have $(\operatorname{Hess} m_k^F(x_k) - \operatorname{Hess} f(x_k))[\xi_k] = \mathbf{P}_{x_k}((H_k - \nabla^2 f(x_k))[\xi_k])$, and thus it holds that

$$\begin{aligned} &\frac{\left| \left\langle \xi_k, (\operatorname{Hess} m_k^F(x_k) - \operatorname{Hess} f(x_k))[\xi_k] \right\rangle_{x_k} \right|}{\|\xi_k\|_{x_k}^2} \\ &\qquad \leq c_1 \frac{\|(H_k - \nabla^2 f(x_k))[z_k - x_k]\|}{\|z_k - x_k\|} \frac{\|z_k - x_k\|}{\alpha_k \|\xi_k\|_{x_k}} + c_2 \frac{\|z_k - x_k - \alpha \xi_k\|}{\alpha_k \|\xi_k\|_{x_k}}, \end{aligned} \tag{4.15}$$

where $c_1, c_2 > 0$ are suitable constants that only depend on $\underline{\varpi}$, $\overline{\varpi}$, κ_H, and κ_F. By (B.4), (4.13), and (4.14), the last term converges to zero as $k \to \infty$. Consequently, we can infer $\langle \xi_k, \operatorname{Hess} m_k(x_k)[\xi_k] \rangle_{x_k} \geq \frac{\nu + \sigma_k}{2} \|\xi_k\|_{x_k}^2$ for all k sufficiently large. This also implies that Algorithm 1 does not stop in iteration $i = 0$. Hence, applying Lemma 4 (ii), we obtain

$$\begin{aligned} \langle \operatorname{grad} m_k(x_k), \xi_k \rangle_{x_k} &\leq \breve{m}_k(\eta_1) - f(x_k) - \frac{1}{2} \langle \xi_k, \operatorname{Hess} m_k(x_k)[\xi_k] \rangle_{x_k} \\ &\leq -\frac{1}{2} \left(\frac{\|g_k\|_{x_k}^4}{\langle g_k, \operatorname{Hess} m_k(x_k)[g_k] \rangle_{x_k}} + \frac{\nu + \sigma_k}{2} \|\xi_k\|_{x_k}^2 \right) \leq -\frac{\nu + \sigma_k}{4} \|\xi_k\|_{x_k}^2, \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $g_k := \operatorname{grad} f(x_k)$. Using this estimate in the proof of Lemma 6, we can now derive a more refined bound for the step size α_k. In particular, it holds that

$$m_k(R_{x_k}(t\xi_k)) \leq -\frac{\rho\nu}{4} t \|\xi_k\|_{x_k}^2, \quad \forall\, t \in [0, \bar{t}], \text{ with } \bar{t} := \frac{1-\rho}{2\overline{\omega}} \min\left\{ \frac{\nu}{\kappa_2\kappa_g + \kappa_1^2\kappa_H}, \frac{1}{\kappa_1^2} \right\},$$

and thus we have

$$-m_k(z_k) \geq \frac{\rho\nu\delta}{4\alpha_0}\bar{t} \cdot \|\xi_k\|_{x_k}^2 \geq \frac{\rho\nu\delta\bar{t}}{4\alpha_0\kappa_1\overline{\omega}} \cdot \|z_k - x_k\|^2 =: \bar{\delta} \cdot \|z_k - x_k\|^2 \tag{4.16}$$

for all k sufficiently large. Next, applying a second-order Taylor expansion, it follows that

$$f(z_k) - f(x_k) - m_k(z_k) \leq \frac{1}{2}\left\langle (\nabla^2 f(x_k^\delta) - H_k)[z_k - x_k], z_k - x_k \right\rangle$$

for some suitable $\delta_k \in [0,1]$ and $x_k^\delta := x_k + \delta_k(z_k - x_k)$. Using the continuity of $\nabla^2 f$, (B.4), and the bound (4.16), we finally obtain

$$1 - \rho_k \leq \frac{1}{2\bar{\delta}}\left[\frac{\|(\nabla^2 f(x_k) - H_k)[z_k - x_k]\|}{\|z_k - x_k\|} + \|\nabla^2 f(x_k^\delta) - \nabla^2 f(x_k)\| \right] \to 0$$

as $k \to \infty$. This finishes the proof. ∎

Next, we establish superlinear convergence of the proposed method. In comparison to Lemma 15, we need a stronger assumption on the matrices H_k to guarantee that the CG method eventually only uses the natural stopping criterion in step S4. In the following, let $\hat{f}_x := f \circ R_x$ denote the pullback of f through R_x at x, and let 0_x be the zero element of $\mathcal{T}_x\mathcal{M}$.

THEOREM 16. *Suppose that Assumption* 13 *and the conditions* (B.1)–(B.3) *and* (B.5) *are satisfied, and let* $\alpha_0 = 1$ *and* $\rho \in (0, \frac{1}{2})$. *Then the sequence* $\{x_k\}$ *converges q-superlinearly to* x_*.

Proof. For convenience, we again set $g_k := \operatorname{grad} f(x_k)$. We further note that the conditions (B.1) and (B.5) imply (A.2)–(A.5). Due to Assumption 13 and applying [7, Proposition 19], the following bound holds for any smooth function h on $\mathcal{M}$:

$$\|\operatorname{Hess} h(x) - \operatorname{Hess} \hat{h}_x(0_x)\|_x \leq \beta_R \|\operatorname{grad} h(x)\|_x, \tag{4.17}$$

where the operator norm is induced by the Riemannian metric on $\mathcal{T}_x\mathcal{M}$. Similar to the proof of Lemma 15 and using (B.3), (B.5), and the uniform estimate

$$|\left\langle \xi, (\operatorname{Hess} m_k^F(x_k) - \operatorname{Hess} f(x_k))[\xi] \right\rangle_{x_k}| \leq c \cdot \|H_k - \nabla^2 f(x_k)\| \|\xi\|_{x_k}^2 \tag{4.18}$$

for $\xi \in \mathcal{T}_{x_k}\mathcal{M}$ and for some constant $c > 0$, we can infer that $\operatorname{Hess} m_k(x_k)$ is positive definite for all k sufficiently large. Thus, the structure of Algorithm 1 now implies that

$$\|g_k + \operatorname{Hess} m_k(x_k)[\xi_k]\|_{x_k} \leq \|g_k\|_{x_k}^\theta, \quad \theta > 1. \tag{4.19}$$

Also, by Lemma 15, we have $\sigma_k \to 0$ as $k \to \infty$. Hence, there exists $\bar{\sigma}$ such that $\sigma_k \leq \bar{\sigma}$ for all $k \in \mathbb{N}$. We next show that the full step size $\alpha_k = 1$ satisfies the Armijo condition (3.9) whenever k is sufficiently large. First, by Lemma 4 (iii) and Remark 3, we have

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

$$\|\xi_k\|_{x_k} \geq \|\eta_1\|_{x_k} = \frac{\|g_k\|_{x_k}^3}{\langle g_k, \operatorname{Hess} m_k(x_k)[g_k]\rangle_{x_k}} \geq \frac{\|g_k\|_{x_k}}{\kappa_{x_k}^M + \sigma_k} \geq \frac{\|g_k\|_{x_k}}{\bar{\kappa}_M + \bar{\sigma}}, \tag{4.20}$$

where $\bar{\kappa}_M$ is defined in Lemma 9. Let $m_k^P := [\hat{m}_k]_{x_k} = m_k \circ R_{x_k}$ denote the pullback of the model function m_k. Combining (4.17), (4.19), and (4.20), it holds that

$$\begin{aligned}
&\|g_k + \operatorname{Hess} m_k^P(0_{x_k})[\xi_k]\|_{x_k} \\
&\qquad \leq \|(\operatorname{Hess} m_k^P(0_{x_k}) - \operatorname{Hess} m_k(x_k))[\xi_k]\|_{x_k} + \|g_k + \operatorname{Hess} m_k(x_k)[\xi_k]\|_{x_k} \\
&\qquad \leq \beta_R \|g_k\|_{x_k} \|\xi_k\|_{x_k} + \|g_k\|_{x_k}^\theta \leq \underbrace{(\beta_R \|g_k\|_{x_k} + (\bar{\kappa}_M + \bar{\sigma})\|g_k\|_{x_k}^{\theta-1})}_{=: \mathcal{C}_k(g_k)} \|\xi_k\|_{x_k}.
\end{aligned}$$

Similar to [32, Proposition 5] and applying a second-order Taylor expansion, it holds that

$$\begin{aligned}
m_k^P(\xi_k) - m_k^P(0_k) - \frac{1}{2}\langle g_k, \xi_k\rangle_{x_k} &= \frac{1}{2}\left\langle g_k + \operatorname{Hess} m_k^P(\delta_k \xi_k)[\xi_k], \xi_k \right\rangle_{x_k} \\
&\leq \left[\mathcal{C}_k(g_k) + \|\operatorname{Hess} m_k^P(\delta_k \xi_k) - \operatorname{Hess} m_k^P(0_{x_k})\|_{x_k}\right] \|\xi_k\|_{x_k}^2 = o(\|\xi_k\|_{x_k}^2),
\end{aligned}$$

where $\delta_k \in [0,1]$ is a suitable constant and we used the last estimate, $\mathcal{C}_k(g_k) \to 0$, and the continuity of the Hessian $\operatorname{Hess} m_k^P$. Therefore, due to $\rho < 0.5$ and $\alpha_0 = 1$, the full step size $\alpha_k = 1$ is chosen in (3.8) if k is sufficiently large and we have $x_{k+1} = R_{x_k}(\xi_k)$. The remaining part of the proof now essentially follows [2, Theorem 7.4.11] and [31, section 4.2.2]. In particular, calculating a first-order Taylor expansion of the pullback gradient $\operatorname{grad} \hat{f}_{x_k}$ and using $\operatorname{grad} \hat{f}_{x_k}(0_{x_k}) = g_k$, the continuity of the pullback Hessian $\operatorname{Hess} \hat{f}_{x_k}$, (4.19), (4.17), (4.18), (B.5), $\sigma_k \to 0$, and (4.12), we obtain

$$\begin{aligned}
&\|\operatorname{grad} \hat{f}_{x_k}(\xi_k)\|_{x_k} \\
&\quad \leq \|\operatorname{grad} \hat{f}_{x_k}(\xi_k) - g_k - \operatorname{Hess} \hat{f}_{x_k}(0_{x_k})[\xi_k]\|_{x_k} + \|g_k + \operatorname{Hess} m_k(x_k)[\xi_k]\|_{x_k} \\
&\qquad + \|(\operatorname{Hess} \hat{f}_{x_k}(0_{x_k}) - \operatorname{Hess} f(x_k))[\xi_k]\|_{x_k} + \|(\operatorname{Hess} f(x_k) - \operatorname{Hess} m_k(x_k))[\xi_k]\|_{x_k} \\
&\quad \leq \|\operatorname{Hess} \hat{f}_{x_k}(\tilde{\delta}_k \xi_k) - \operatorname{Hess} \hat{f}_{x_k}(0_{x_k})\|_{x_k} \|\xi_k\|_{x_k} + \|g_k\|_{x_k}^\theta + \beta_R \|g_k\|_{x_k} \|\xi_k\|_{x_k} \\
&\qquad + c \cdot \|H_k - \nabla^2 f(x_k)\| \|\xi_k\|_{x_k} + \sigma_k \|\xi_k\|_{x_k} \\
&\quad = o(\|g_k\|_{x_k}),
\end{aligned}$$

where $\tilde{\delta}_k \in [0,1]$ is again an appropriate constant. By [2, Lemma 7.4.9], this implies that

$$\frac{\|\operatorname{grad} f(x_{k+1})\|_{x_{k+1}}}{\|\operatorname{grad} f(x_k)\|_{x_k}} \leq \tilde{c} \cdot \frac{\|\operatorname{grad} \hat{f}_{x_k}(\xi_k)\|_{x_k}}{\|g_k\|_{x_k}} \to 0 \quad \text{as} \quad k \to \infty \tag{4.21}$$

for some $\tilde{c} > 0$. Moreover, since the Hessian $\operatorname{Hess} f(x_*)$ is positive definite, [2, Lemma 7.4.8] and (4.21) further imply that

$$\frac{\operatorname{dist}(x_{k+1}, x_*)}{\operatorname{dist}(x_k, x_*)} \to 0$$

as $k \to \infty$. (Here, $\operatorname{dist}(\cdot,\cdot)$ denotes the Riemannian geodesic distance; see [2].) ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

5. Numerical results. In this section, we test a variety of examples to illustrate the efficiency of our adaptively regularized Newton method (ARNT).[1] We mainly compare Algorithm 2 with the Riemannian gradient method using the BB step size for initialization (GBB), described in subsection 2.1, and the Riemannian trust region method (RTR) Manopt. All codes are written in MATLAB. Note that Huang et al. [21] implement a C-language version of RTR to further accelerate the method. The efficiency of ARNT can also be improved in a similar way. All experiments were performed on a workstation with Intel Xenon E5-2680 v3 processors at 2.50 GHz (×12) and 128 GB memory running CentOS 6.8 and MATLAB R2015b.

The default values of the GBB parameters are set to $\rho = 10^{-4}, \delta = 0.2$, and $\varrho = 0.85$. We have extensively tuned the stopping criterion of the truncated CG method implemented in RTR and found that adding a rule $\|r_{j+1}\| \leq \min\{0.1, 0.1\|r_0\|\}$ often improves the performance of RTR. All other default settings of RTR were used. For ARNT, we set $\eta_1 = 0.01$, $\eta_2 = 0.9$, $\gamma_0 = 0.2$, $\gamma_1 = 1$, $\gamma_2 = 10$, and $\sigma_k = \hat{\sigma}_k \|\mathrm{grad}\, f(x_k)\|$, where $\hat{\sigma}_k$ is updated by (3.12) with $\hat{\sigma}_0 = 10$. The parameters in Algorithm 1 are chosen as follows: $\rho = 10^{-4}$, $\delta = 0.2$, $\theta = 1$, and $T = 0.1$. Furthermore, when an estimation of the absolute value of the negative curvature, denoted by σ_{est}, is available at the kth subproblem (see step S2 in Algorithm 1), we calculate

$$\sigma_{k+1}^{new} = \max\{\sigma_{k+1}, \sigma_{est} + \tilde{\gamma}\}$$

with some small $\tilde{\gamma} \geq 0$. Then the parameter σ_{k+1} is reset to σ_{k+1}^{new}. This change does not affect our convergence results. For fair comparisons, all algorithms are stopped when the norm of the Riemannian gradient is less than 10^{-6} unless a different tolerance is specified. The algorithms also terminate if a maximum number of iterations is reached. We use a maximum number of 10^4 iterations in GBB and 500 in ARNT and RTR. In the implementation of ARNT and RTR, the GBB method is used to obtain a better initial point. Here, GBB is run with stopping criterion $\|\mathrm{grad}\, f(x_k)\| \leq 10^{-3}$ and a maximum of 2000 iterations. The maximum number of inner iterations in ARNT is chosen adaptively depending on the norm of the Riemannian gradient.

In the subsequent tables, the column "its" represents the total number of iterations in GBB, while the two numbers of the column "its" in ARNT and RTR are the number of outer iterations and the average numbers of inner iterations. The columns "f," "nrmG," and "time" denote the final objective value, the final norm of the Riemannian gradient, and the CPU time that the algorithms spent to reach the stopping criterions, respectively.

5.1. Low rank nearest correlation matrix estimation. Given a symmetric matrix C and a nonnegative symmetric weight matrix H, the low rank nearest correlation matrix problem is given as

$$\min_{X \in \mathbb{R}^{n\times n}} \frac{1}{2}\|H \odot (X - C)\|_F^2, \quad \text{s.t.} \quad X_{ii} = 1, \quad \mathrm{rank}(X) \leq p, \quad X \succeq 0, \tag{5.1}$$

for all $i = 1, \ldots, n$ and for $p \leq n$. By expressing $X = V^\top V$ with $V = [V_1, \ldots, V_n] \in \mathbb{R}^{p\times n}$, problem (5.1) can be converted into

$$\min_{V \in \mathbb{R}^{p\times n}} \frac{1}{2}\|H \odot (V^\top V - C)\|_F^2, \quad \text{s.t.} \quad \|V_i\|_2 = 1, \quad i = 1, \ldots, n.$$

[1] Downloadable at https://github.com/wenstone/ARNT.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 1
Numerical results of Example 1 *on low rank nearest correlation estimation.*

	GBB			AdaGBB			ARNT			RTR		
p	its	nrmG	time	its	nrmG	time	its	nrmG	time	its	nrmG	time
$H = \mathbf{1}$, $n = 500$												
5	207	3.5e-7	1.2	227	8.8e-7	0.9	24(14)	1.2e-7	1.3	31(8)	2.7e-7	1.2
10	173	8.7e-7	0.5	215	9.6e-7	0.5	11(11)	3.2e-7	0.6	11(12)	6.7e-7	0.7
20	293	5.3e-7	0.9	352	6.3e-7	1.1	13(18)	1.2e-7	0.9	12(21)	8.4e-7	1.0
50	2622	1.0e-6	9.4	1306	8.6e-7	5.8	43(37)	2.4e-7	5.1	39(20)	5.5e-7	3.0
100	3286	9.0e-7	17.4	3614	9.9e-7	13.6	52(51)	6.0e-7	11.1	52(30)	3.6e-7	6.8
150	9358	9.9e-7	47.4	10000	3.4e-6	62.5	51(75)	1.6e-7	18.7	55(54)	5.1e-7	13.8
200	10000	2.8e-5	82.1	10000	2.1e-4	46.7	70(70)	5.6e-7	31.0	77(49)	9.2e-7	18.5
$H \neq \mathbf{1}$, $n = 500$												
5	1016	9.3e-7	5.1	744	9.3e-7	3.4	115(19)	1.9e-7	6.0	290(21)	3.5e-7	20.4
10	722	1.0e-6	3.3	431	5.6e-7	1.3	40(61)	4.9e-7	6.3	28(40)	6.8e-7	3.6
20	923	7.8e-7	3.1	715	4.1e-7	4.8	20(70)	8.2e-7	4.4	23(52)	7.1e-7	3.9
50	10000	1.6e+0	36.8	10000	3.1e-6	65.3	69(105)	6.0e-7	24.0	116(115)	7.0e-7	50.3
100	10000	1.2e-1	47.0	10000	4.5e-2	67.6	345(119)	5.0e-7	154.8	449(169)	9.7e-7	331.2
150	10000	3.6e-1	49.5	10000	5.4e-2	43.8	500(119)	1.4e-1	269.5	500(168)	5.9e-1	385.9
200	10000	8.3e-2	65.5	10000	6.5e-2	47.8	500(125)	7.0e-2	341.1	500(165)	2.0e-1	414.1

In this subsection, we also use a version of the Adagrad method (2.6) in our numerical comparison. It is dubbed as AdaGBB because its setting is similar to GBB. We select a few typical test problems as follows.

Example 1. Let $n = 500$, and let $C_{ij} = 0.5 + e^{-0.05|i-j|}$ for $i, j = 1, \ldots, n$. The weight matrix H is either $\mathbf{1}$ or a random matrix whose entries are mostly uniformly distributed in $[0.1, 10]$ except that 200 entries are distributed in $[0.01, 100]$.

Example 2. The matrix C is obtained from the real gene correlation matrix Leukemia. The weight matrix H is either $\mathbf{1}$ or a random matrix whose entries are set as in Example 1. Results for other data sets, such as Lymph, ER, and Hereditarybc, are not reported here due to their similarity.

Example 3. Let $n = 943$. The matrix C is based on $100,000$ ratings for 1682 movies by 943 users from the Movielens data sets. The weight matrix H is provided by T. Fushiki at the Institute of Statistical Mathematics, Japan.

The detailed numerical results are reported in Tables 1 and 2. For Example 1, all methods perform well if p is small. For the cases with $H \neq \mathbf{1}$, ARNT is the best when $p = 50$ and $p = 100$, while all of them fail when $p = 150$ and $p = 200$. For Example 2, GBB may not converge when p is large, and ARNT is efficient whenever p is small or large. In particular, ARNT is better than RTR on Leukemia with $H \neq \mathbf{1}$, and RTR may fail on a few instances. For Example 3, we can see that GBB and RTR fail to converge when p is small, while ARNT, and AdaGBB still work. In fact, we observe negative curvatures of the Hessian at many iterations of ARNT, and the strategy (3.7) indeed helps the convergence.

5.2. Kohn–Sham total energy minimization. Using a suitable discretization scheme, we can formulate a finite dimensional approximation to the continuous Kohn–Sham (KS) minimization problem [38] as

$$\min_{X \in \mathbb{C}^{n \times p}} \quad f(X) \quad \text{s.t.} \quad X^* X = I,$$

where $f(X) := \frac{1}{4}\mathrm{tr}(X^* L X) + \frac{1}{2}\mathrm{tr}(X^* V_{ion} X) + \frac{1}{2}\sum_i \sum_l |x_i^* \omega_l| + \frac{1}{4}\rho L^\dagger + \frac{1}{2} e^\top \epsilon_{xc}(\rho)$, $X = [x_1, \ldots, x_p] \in \mathbb{C}^{n \times p}$, $\rho(X) := \mathrm{diag}(XX^*)$, L is a finite dimensional Laplacian

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 2
Numerical results of Example 2 *on low rank nearest correlation estimation (continued).*

	GBB			AdaGBB			ARNT			RTR		
p	its	nrmG	time	its	nrmG	time	its	nrmG	time	its	nrmG	time
$H = \mathbf{1}$ (Leukemia, $n = 1255$)												
5	272	8.9e-7	4.2	261	4.8e-7	2.9	15(16)	4.0e-7	5.6	23(9)	4.0e-7	5.2
10	540	9.6e-7	12.5	453	8.2e-7	5.7	23(20)	5.6e-7	8.4	48(21)	6.8e-7	13.8
20	1064	9.6e-7	23.0	1602	1.0e-6	26.2	34(31)	4.1e-7	14.8	131(25)	1.5e-7	39.4
50	1917	8.5e-7	32.9	2535	4.0e-7	35.8	33(49)	2.6e-7	26.0	28(42)	1.6e-7	17.4
100	10000	4.1e-5	169.4	10000	2.4e-5	156.9	35(27)	9.6e-7	19.5	35(28)	5.6e-7	18.0
150	10000	3.1e-5	194.8	10000	9.2e-5	184.4	40(25)	4.8e-7	24.4	36(27)	9.5e-7	20.0
200	10000	3.2e-5	232.1	10000	3.9e-4	200.1	37(25)	5.0e-7	24.8	36(27)	4.6e-7	22.5
$H \neq \mathbf{1}$ (Leukemia, $n = 1255$)												
5	1404	5.9e-7	55.0	762	5.5e-7	13.3	44(20)	3.8e-7	16.3	500(16)	3.7e-3	137.7
10	680	9.7e-7	21.8	608	9.8e-7	13.0	23(22)	9.5e-7	11.2	500(20)	2.1e-3	169.5
20	2461	9.4e-7	77.1	2250	9.3e-7	51.6	59(32)	9.5e-7	31.0	500(34)	3.5e-4	289.0
50	3354	9.7e-7	82.0	1790	7.8e-7	47.1	33(86)	4.8e-7	48.2	58(74)	1.2e-7	79.2
100	10000	1.8e-2	170.5	10000	1.8e-3	158.6	36(51)	7.7e-7	37.5	44(53)	4.9e-7	48.7
150	10000	3.4e-3	194.9	10000	2.1e-3	197.8	43(52)	4.8e-7	48.2	51(52)	4.8e-7	57.1
200	10000	3.9e-3	216.5	10000	4.3e-2	205.8	46(50)	4.6e-7	55.9	50(52)	6.5e-7	65.4

TABLE 3
Numerical results of Example 3 *on low rank nearest correlation estimation.*

	GBB			AdaGBB			ARNT			RTR		
p	its	nrmG	time	its	nrmG	time	its	nrmG	time	its	nrmG	time
5	10000	1.7e+02	196.5	4178	6.9e-7	41.3	260(8)	9.4e-7	38.1	500(12)	8.8e-2	78.4
10	10000	3.0e-4	207.4	4973	8.2e-7	103.8	347(12)	8.4e-7	58.9	500(17)	9.3e-2	102.5
20	10000	1.5e-4	198.3	5089	7.1e-7	86.6	237(24)	8.3e-7	63.4	500(23)	9.7e-2	152.0
50	10000	9.1e-5	288.1	3675	1.0e-6	90.2	34(58)	2.0e-7	38.1	63(82)	7.7e-7	80.2
100	10000	3.6e-4	181.6	10000	2.5e-6	258.0	26(118)	7.1e-7	50.1	19(428)	7.1e-7	120.4
150	10000	3.5e-2	124.2	10000	4.4e-5	241.7	35(134)	3.0e-7	76.1	18(688)	9.0e-7	173.2
200	10000	3.5e-2	153.7	10000	7.2e-5	245.3	37(130)	5.5e-7	78.4	16(758)	8.3e-7	162.0

operator, V_{ion} corresponds to the ionic pseudopotentials, w_l represents a discretized pseudopotential reference projection function, and ϵ_{xc} is related to the exchange correlation energy.

Our experiments are based on the KSSOLV package [41]. As in [38], we use the Wirtinger calculus [27] to compute the complex gradient and Hessian of the function f. Let us also note that the Lipschitz continuity required in assumption (A.1) may not be satisfied for all types of exchange correlations. However, for the correlation that is defined by the Perdew–Zunger formula and used in this example, Lipschitz continuity was established in [36, Lemma 3.3]. In addition to GBB and RTR, we further compare ARNT with the self-consistent field (SCF) iteration and the regularized trust-region method TRQH in [38]. In the implementation of TRQH, RTR, and ARNT, we use the same initial point obtained by GBB. Note that TRQH essentially coincides with ARNT except that the subproblem (3.4) is solved by GBB.

A summary of the computational results is given in Table 4. All algorithms reach the same objective function value when the gradient norm criterion is satisfied. ARNT and RTR take a small number of outer iterations to converge and often exhibit a fast convergence rate. In particular, ARNT tends to be more efficient than other algorithms on "graphene30" and "qdot." It can be even faster than SCF when SCF works well. ARNT also outperforms TRQH. This shows that the accuracy of solving the subproblem (3.4) is indeed important.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 4
Numerical results on KS total energy minimization.

solver	f	its	nrmG	time	f	its	nrmG	time
	alanine				al			
SCF	-6.1162e+01	14	3.9e-7	25.0	-1.5784e+01	101	4.5e-2	146.9
OptM	-6.1162e+01	80	7.1e-7	25.5	-1.5804e+01	1461	9.9e-7	391.1
TRQH	-6.1162e+01	6(16)	6.5e-7	39.7	-1.5804e+01	39(16)	9.6e-7	411.9
ARNT	-6.1162e+01	3(9)	3.9e-7	24.4	-1.5804e+01	5(113)	3.5e-7	196.4
RTR	-6.1162e+01	3(9)	4.1e-7	24.3	-1.5804e+01	5(108)	9.9e-8	188.9
	benzene				c12h26			
SCF	-3.7226e+01	13	4.0e-7	14.3	-8.1536e+01	13	9.1e-7	30.2
OptM	-3.7226e+01	68	5.1e-7	13.4	-8.1536e+01	89	8.8e-7	34.1
TRQH	-3.7226e+01	6(12)	9.3e-7	19.2	-8.1536e+01	7(12)	9.7e-7	50.0
ARNT	-3.7226e+01	3(10)	9.2e-8	13.3	-8.1536e+01	3(13)	6.4e-7	29.5
RTR	-3.7226e+01	3(10)	8.1e-8	13.6	-8.1536e+01	3(13)	5.2e-7	29.5
	c2h6				co2			
SCF	-1.4420e+01	10	6.8e-7	2.5	-3.5124e+01	10	3.1e-7	2.6
OptM	-1.4420e+01	59	9.1e-7	2.6	-3.5124e+01	59	5.2e-7	2.6
TRQH	-1.4420e+01	6(12)	8.7e-7	4.0	-3.5124e+01	6(12)	3.7e-7	3.9
ARNT	-1.4420e+01	3(8)	4.7e-7	2.5	-3.5124e+01	3(9)	3.1e-7	2.5
RTR	-1.4420e+01	3(7)	3.9e-7	2.7	-3.5124e+01	3(10)	2.5e-7	2.7
	ctube661				graphene16			
SCF	-1.3464e+02	16	3.1e-7	88.5	-9.4028e+01	101	5.8e-4	160.0
OptM	-1.3464e+02	101	7.2e-7	93.0	-9.4046e+01	187	8.5e-7	40.8
TRQH	-1.3464e+02	6(19)	3.2e-7	138.5	-9.4046e+01	8(19)	9.5e-7	70.3
ARNT	-1.3464e+02	3(11)	4.9e-7	78.3	-9.4046e+01	3(19)	8.6e-7	40.3
RTR	-1.3464e+02	3(11)	4.2e-7	78.2	-9.4046e+01	3(19)	7.3e-7	40.7
	graphene30				h2o			
SCF	-1.7358e+02	101	2.2e-3	860.6	-1.6441e+01	9	1.4e-7	1.8
OptM	-1.7360e+02	378	6.5e-7	517.0	-1.6441e+01	58	8.9e-7	2.0
TRQH	-1.7360e+02	12(38)	8.6e-7	783.9	-1.6441e+01	5(38)	8.4e-7	2.9
ARNT	-1.7360e+02	4(33)	2.5e-7	446.8	-1.6441e+01	3(11)	3.9e-7	1.8
RTR	-1.7360e+02	100(4)	2.3e-5	828.8	-1.6441e+01	3(11)	3.1e-7	2.1
	hnco				nic			
SCF	-2.8635e+01	12	3.5e-7	3.3	-2.3544e+01	10	7.2e-7	1.2
OptM	-2.8635e+01	131	9.7e-7	5.6	-2.3544e+01	63	9.9e-7	1.1
TRQH	-2.8635e+01	7(21)	9.5e-7	6.9	-2.3544e+01	8(21)	9.3e-7	2.3
ARNT	-2.8635e+01	3(15)	7.5e-7	3.7	-2.3544e+01	3(8)	4.4e-7	1.0
RTR	-2.8635e+01	3(16)	7.7e-7	4.5	-2.3544e+01	3(8)	4.6e-7	1.3
	ptnio				qdot			
SCF	-2.2679e+02	66	7.7e-7	146.2	2.7702e+01	101	3.4e-2	22.3
OptM	-2.2679e+02	495	5.3e-7	145.6	2.7695e+01	2000	3.3e-6	70.8
TRQH	-2.2679e+02	23(39)	9.3e-7	286.0	2.7695e+01	91(39)	9.9e-7	115.8
ARNT	-2.2679e+02	4(52)	6.9e-7	132.4	2.7695e+01	27(65)	7.1e-7	64.5
RTR	-2.2679e+02	4(46)	8.5e-7	122.5	2.7695e+01	37(68)	4.0e-7	83.3

5.3. Bose–Einstein condensates. The total energy in the Bose–Einstein condensates (BEC) is defined as

$$E(\psi) = \int_{\mathbb{R}^d} \left[\frac{1}{2}|\nabla\psi(\mathbf{x})|^2 + V(\mathbf{x})|\psi(\mathbf{x})|^2 + \frac{\beta}{2}|\psi(\mathbf{x})|^4 - \Omega\bar{\psi}(\mathbf{x})L_z(\mathbf{x}) \right] d\mathbf{x},$$

where $\mathbf{x} \in \mathbb{R}^d$ is the spatial coordinate vector, $\bar{\psi}$ denotes the complex conjugate of ψ, $L_z = -i(x\partial - y\partial x)$, $V(x)$ is an external trapping potential, and β, Ω are given constants. Using a suitable discretization, such as finite differences or the sine pseudospectral and Fourier pseudospectral (FP) method, we can reformulate the BEC problem as follows:

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 5
Numerical results on BEC with the potential function $V(x, y)$.

solver	f	its	nrmG	time	f	its	nrmG	time
β =500								
	$\Omega = 0.00$				$\Omega = 0.25$			
OptM	8.5118	58	6.6e-5	1.4	8.5106	103	9.7e-5	12.3
TRQH	8.5118	4(17)	1.5e-4	2.0	8.5106	5(22)	1.9e-4	21.9
ARNT	8.5118	3(24)	1.2e-5	1.5	8.5106	4(53)	1.6e-5	17.7
RTR	8.5118	3(25)	1.3e-5	1.5	8.5106	3(23)	6.0e-5	15.1
	$\Omega = 0.50$				$\Omega = 0.60$			
OptM	8.0246	276	9.0e-5	32.3	7.5890	301	1.0e-4	19.9
TRQH	8.0246	5(53)	2.0e-4	60.7	7.5890	5(60)	1.9e-4	35.4
ARNT	8.0197	3(62)	6.5e-5	21.3	7.5890	3(67)	5.7e-5	22.1
RTR	8.0246	11(113)	1.0e-4	56.5	7.5890	3(61)	5.2e-5	23.8
	$\Omega = 0.70$				$\Omega = 0.80$			
OptM	6.9731	340	1.0e-4	56.3	6.1016	386	1.0e-4	65.2
TRQH	6.9731	7(55)	2.0e-4	61.6	6.1016	5(64)	2.0e-4	83.1
ARNT	6.9731	10(99)	8.7e-5	44.4	6.1016	10(104)	8.7e-5	70.6
RTR	6.9731	99(118)	9.3e-5	234.2	6.1016	18(130)	7.7e-5	130.1
	$\Omega = 0.90$				$\Omega = 0.95$			
OptM	4.7784	10000	1.2e-3	243.6	3.7419	10000	7.4e-4	241.6
TRQH	4.7778	277(176)	2.0e-4	1090.9	3.7416	363(181)	2.0e-4	1185.1
ARNT	4.7777	147(132)	9.6e-5	413.3	3.7414	500(147)	2.6e-4	1204.0
RTR	4.7777	500(147)	8.5e-4	1250.4	3.7415	500(172)	9.7e-4	1419.0
β =1000								
	$\Omega = 0.00$				$\Omega = 0.25$			
OptM	11.9718	76	4.6e-5	3.0	11.9266	358	9.9e-5	40.2
TRQH	11.9718	4(15)	1.0e-4	1.5	11.9266	4(50)	1.7e-4	44.3
ARNT	11.9718	3(16)	3.1e-5	0.9	11.9266	15(70)	2.5e-5	40.9
RTR	11.9718	3(16)	3.8e-5	0.8	11.9266	15(70)	8.7e-5	46.4
	$\Omega = 0.50$				$\Omega = 0.60$			
OptM	11.1054	396	1.0e-4	32.6	10.4392	5524	1.0e-4	140.4
TRQH	11.1326	6(53)	2.0e-4	36.3	10.4437	9(98)	2.0e-4	92.8
ARNT	11.1326	20(66)	5.9e-5	36.8	10.4392	20(73)	7.6e-5	77.9
RTR	11.1326	32(78)	5.8e-5	68.9	10.4392	93(80)	9.8e-5	187.6
	$\Omega = 0.70$				$\Omega = 0.80$			
OptM	9.5283	990	1.0e-4	63.7	8.2627	10000	5.5e-4	231.9
TRQH	9.5301	102(156)	2.0e-4	404.1	8.2610	453(177)	2.0e-4	1427.0
ARNT	9.5301	60(81)	9.3e-5	140.4	8.2610	202(105)	6.7e-5	412.7
RTR	9.5301	293(91)	8.6e-5	478.8	8.2610	500(113)	5.5e-4	972.7
	$\Omega = 0.90$				$\Omega = 0.95$			
OptM	6.3611	10000	3.0e-3	230.8	4.8856	10000	5.2e-4	241.4
TRQH	6.3607	142(170)	2.0e-4	595.6	4.8831	172(178)	2.0e-4	708.1
ARNT	6.3607	500(110)	2.8e-3	931.5	4.8822	500(121)	1.5e-3	1015.8
RTR	6.3607	500(122)	7.6e-4	1010.8	4.8823	500(137)	1.9e-3	1103.8

$$\min_{x\in\mathbb{C}^M} \; f(x) := \frac{1}{2}x^*Ax + \frac{\beta}{2}\sum_{j=1}^{M}|x_j|^4, \qquad \text{s.t.} \quad \|x\|_2 = 1,$$

where $M \in \mathbb{N}$, β is a given real constant, and $A \in \mathbb{C}^{M\times M}$ is a Hermitian matrix.

In this numerical experiment, we again use the Wirtinger calculus to calculate the complex gradient and Hessian of the objective function. We stop GBB, ARNT, RTR, and TRQH (the Newton method in [40]) when the gradient norm is less than 10^{-4} or the maximum number of iterations is reached. For TRQH, the stopping criterion $\|x^{k+1} - x^k\|_\infty \le \epsilon_x$ is added for some small constant ϵ_x since TRQH often does not converge under the gradient norm criterion. We take $d = 2$ and test two different

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 6
Numerical results on low rank matrix completion with the fixed $k = 10, r_S = 0.8$ but different n.

	GBB			ARNT			RTR		
n	its	nrmG	time	its	nrmG	time	its	nrmG	time
1000	603	5.1e-7	12.5	6(84)	3.4e-7	7.7	8(91)	6.6e-7	8.2
2000	570	9.2e-7	43.9	5(72)	8.9e-7	23.6	8(86)	6.2e-7	28.2
4000	671	9.7e-7	179.8	6(82)	4.6e-7	94.8	9(85)	2.0e-7	104.8
8000	666	9.8e-7	694.2	5(104)	5.2e-7	320.1	8(130)	5.4e-7	394.5

TABLE 7
Numerical results on low rank matrix completion with fixed $n = 4000, r_S = 0.95$ but different k.

	GBB			ARNT			RTR		
k	its	nrmG	time	its	nrmG	time	its	nrmG	time
10	5252	1.0e-6	1415.9	13(133)	7.4e-7	392.1	12(236)	4.4e-7	438.2
20	2126	1.0e-6	600.8	7(125)	3.9e-7	269.5	9(195)	2.3e-7	315.9
30	1488	1.0e-6	438.8	6(132)	3.1e-7	255.2	9(214)	2.6e-7	329.9
40	1010	9.3e-7	311.4	5(103)	1.1e-7	220.5	5(103)	1.1e-7	219.4
50	1494	7.9e-7	477.1	4(103)	1.5e-7	273.8	4(103)	1.6e-7	272.5
60	1398	9.9e-7	480.4	4(110)	5.7e-7	313.3	4(114)	5.7e-7	315.2

potential functions $V(x,y) = \frac{1}{2}x^2 + \frac{1}{2}y^2$. The BEC problem is discretized by FP on the bounded domain $(-16,16)^2$ with $\beta = 500, 1000$ and different values of Ω ranging from 0 to 0.95. Under the same settings as in [40, section 4.3], we use the mesh refinement procedure with the coarse meshes $(2^4+1)\times(2^4+1), (2^5+1)\times(2^5+1)$, $\ldots, (2^7+1)\times(2^7+1)$ to gradually obtain an initial solution point on the finest mesh $(2^8+1)\times(2^8+1)$. For a fair comparison, all algorithms are tested with mesh refinement and start from the same initial point on the coarsest mesh with $\phi(x,y) = \frac{(1-\Omega)\phi_1(x,y)+\Omega\phi_2(x,y)}{\|(1-\Omega)\phi_1(x,y)+\Omega\phi_2(x,y)\|}$ and $\phi_1(x,y) = \frac{1}{\sqrt{\pi}}e^{-(x^2+y^2)/2}$, $\phi_2(x,y) = \frac{x+iy}{\sqrt{\pi}}e^{-(x^2+y^2)}/2$.

A summary of the results is presented in the Table 5. The parameter ϵ_x for TRQH is set to 10^{-8} in these two cases. The tables show that GBB does not to converge within 10000 steps in several cases. TRQH usually performs worse than ARNT in terms of accuracy and time except in the cases $\beta = 1000$ with $\Omega = 0.95$ in Table 5, where ARNT finds a point with a smaller objective function value. ARNT performs no worse than RTR in most experiments.

5.4. Low rank matrix completion. Given a partially observed matrix $A \in \mathbb{R}^{m\times n}$, we want to find the lowest rank matrix to fit A on the known elements. This problem can be formulated as follows:

$$\min_{X\in\mathbb{R}^{m\times n}} \; f(X) := \frac{1}{2}\|P_\Omega(X) - A\|_F^2 \quad \text{s.t.} \quad X \in \{X \in \mathbb{R}^{m\times n} : \operatorname{rank}(X) = k\},$$

where $P_\Omega : \mathbb{R}^{m\times n} \to \mathbb{R}^{m\times n}$, $P_\Omega(X)_{i,j} := X_{i,j}$ if $(i,j) \in \Omega$, and $P_\Omega(X)_{i,j} := 0$ if $(i,j) \notin \Omega$ is the projection onto Ω and Ω is a subset of $\{1,\ldots,m\}\times\{1,\ldots,n\}$. More details can be found in [37].

Similar to [37], we construct random numerical examples as follows. We first take two Gaussian random matrices $A_L, A_R \in \mathbb{R}^{n\times k}$, then uniformly sample the index set Ω for a given cardinality and set the matrix $A := P_\Omega(A_L A_R^\top)$. Since the degrees of freedom in a nonsymmetric matrix of rank k is given $k(2n-k)$, we define the ratio $r_S = (k(2n-k))^{-1}|\Omega|$. In this example, we only penalize $x - x_k$ on the known set Ω in the implementation of ARNT to reduce the computational costs. (That is, the penalization term in the subproblem (3.3) is set to $\sigma_k\|P_\Omega(x-x_k)\|^2$.) In Tables 6–8,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

TABLE 8
Numerical results on low rank matrix completion with fixed $n = 8000, k = 10$ but different r_S.

	GBB			ARNT			RTR		
r_S	its	nrmG	time	its	nrmG	time	its	nrmG	time
0.1	86	3.7e-7	88.1	3(11)	4.4e-7	54.7	3(11)	4.3e-7	53.2
0.2	89	8.6e-7	93.9	3(14)	3.4e-7	55.5	3(14)	3.4e-7	54.0
0.3	117	9.5e-7	119.7	3(14)	4.2e-7	67.3	3(14)	4.2e-7	66.1
0.5	173	8.5e-7	178.8	3(18)	7.0e-7	111.4	3(18)	7.0e-7	109.7
0.8	666	9.8e-7	700.2	5(104)	5.2e-7	318.7	8(130)	5.4e-7	388.8

we can see that ARNT and RTR perform better than GBB regardless of whether the dimension n and rank k are large or small. We often observe that ARNT tends to outperform RTR when negative curvature is encountered.

6. Conclusions. In this paper, we propose a regularized Newton method for optimization problems on Riemannian manifolds. We use a second-order approximation of the objective function in the Euclidean space to form a sequence of quadratic subproblems while keeping the manifold constraints. A modified Newton method is then developed and analyzed to solve the resulting subproblems. Based on a Steihaug-type CG method, we construct a specific search direction that can use negative curvature information of the Riemannian Hessian. We show that our method enjoys favorable convergence properties and converges with a locally superlinear rate. Numerical experiments are performed on the nearest correlation matrix estimation, KS total energy minimization, BEC, and low rank matrix completion problems. The comparisons illustrate that our proposed method is promising. In particular, it can often reach a certain level of accuracy faster than other state-of-the-art algorithms. Our algorithm performs comparable to the RTR method and usually achieves a better convergence rate once negative curvature is encountered. Finally, let us mention that the performance of our proposed algorithmic framework may be further improved if a more specialized and problem-dependent solver for the inner subproblem is used.

Acknowledgments. We would like to thank Bo Jiang for the helpful discussion on optimization with orthogonality constraints. The authors are grateful to the associate editor and two anonymous referees for their valuable comments and suggestions.

REFERENCES

[1] P.-A. Absil, C. G. Baker, and K. A. Gallivan, *Trust-region methods on Riemannian manifolds*, Found. Comput. Math., 7 (2007), pp. 303–330.
[2] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*, Princeton University Press, Princeton, NJ, 2008.
[3] P.-A. Absil, R. Mahony, and J. Trumpf, *An extrinsic look at the Riemannian Hessian*, in Geometric Science of Information, Springer, New York, 2013, pp. 361–368.
[4] P.-A. Absil and J. Malick, *Projection-like retractions on matrix manifolds*, SIAM J. Optim., 22 (2012), pp. 135–158.
[5] N. Agarwal, Z. Allen-Zhu, B. Bullins, E. Hazan, and T. Ma, *Finding approximate local minima faster than gradient descent*, in Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2017, pp. 1195–1199.
[6] E. Birgin and J. Martinez, *The use of quadratic regularization with a cubic descent condition for unconstrained optimization*, SIAM J. Optim., 27 (2017), pp. 1049–1074.
[7] N. Boumal, P.-A. Absil, and C. Cartis, *Global rates of convergence for nonconvex optimization on manifolds*, IMA J. Numer. Anal., 00 (2018), pp. 1–33.
[8] N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, *Manopt, a Matlab toolbox for optimization on manifolds*, J. Mach. Learn. Res., 15 (2014), pp. 1455–1459, http://www.manopt.org.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

[9] R. H. Byrd, J. Nocedal, and F. Oztoprak, *An inexact successive quadratic approximation method for L-1 regularized optimization*, Math. Program., 157 (2016), pp. 375–396.
[10] Y. Carmon, J. C. Duchi, O. Hinder, and A. Sidford, *Accelerated Methods for Non-Convex Optimization*, preprint, arXiv:1611.00756, 2016.
[11] C. Cartis, N. I. M. Gould, and P. L. Toint, *Adaptive cubic regularisation methods for unconstrained optimization. Part* I*: Motivation, convergence and numerical results*, Math. Program., 127 (2011), pp. 245–295.
[12] H. Chen, Y. Sun, J. Gao, Y. Hu, and B. Yin, *Fast optimization algorithm on Riemannian manifolds and its application in low-rank learning*, Neurocomputing, 291 (2018), pp. 59–70.
[13] F. E. Curtis and D. P. Robinson, *Exploiting Negative Curvature in Deterministic and Stochastic Optimization*, preprint, arXiv:1703.00412v3, 2017.
[14] J. Duchi, E. Hazan, and Y. Singer, *Adaptive subgradient methods for online learning and stochastic optimization*, J. Mach. Learn. Res., 12 (2011), pp. 2121–2159.
[15] A. Edelman, T. A. Arias, and S. T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM J. Matrix Anal. Appl., 20 (1998), pp. 303–353.
[16] D. Gabay, *Minimizing a differentiable function over a differential manifold*, J. Optim. Theory Appl., 37 (1982), pp. 177–219.
[17] B. Gao, X. Liu, X. Chen, and Y. Yuan, *A new first-order framework for orthogonal constrained optimization problems*, SIAM J. Optim., 28 (2018), pp. 302–332.
[18] N. I. M. Gould, S. Lucidi, M. Roma, and P. L. Toint, *Exploiting negative curvature directions in linesearch methods for unconstrained optimization*, Optim. Methods Softw., 14 (2000), pp. 75–98.
[19] W. Huang, P.-A. Absil, and K. A. Gallivan, *A Riemannian symmetric rank-one trust-region method*, Math. Program., 150 (2015), pp. 179–216.
[20] W. Huang, P.-A. Absil, and K. A. Gallivan, *A Riemannian BFGS method for nonconvex optimization problems*, in Numerical Mathematics and Advanced Applications ENUMATH 2015, Springer, New York, 2016, pp. 627–634.
[21] W. Huang, P.-A. Absil, K. A. Gallivan, and P. Hand, *ROPTLIB: An Object-Oriented C++ Library for Optimization on Riemannian Manifolds*, Technical report FSU16-14.v2, Florida State University, Tallahassee, 2016.
[22] W. Huang, K. A. Gallivan, and P.-A. Absil, *A Broyden class of quasi-Newton methods for Riemannian optimization*, SIAM J. Optim., 25 (2015), pp. 1660–1685.
[23] B. Iannazzo and M. Porcelli, *The Riemannian Barzilai-Borwein method with nonmonotone line search and the matrix geometric mean computation*, IMA J. Numer. Anal., 38 (2018), pp. 495–517.
[24] B. Jiang and Y.-H. Dai, *A framework of constraint preserving update schemes for optimization on Stiefel manifold*, Math. Program., 153 (2015), pp. 535–575.
[25] E. W. Karas, S. A. Santos, and B. F. Svaiter, *Algebraic rules for quadratic regularization of Newton method*, Comput. Optim. Appl., 60 (2015), pp. 343–376.
[26] D. Kressner, M. Steinlechner, and B. Vandereycken, *Low-rank tensor completion by Riemannian optimization*, BIT Numer. Math., 54 (2014), pp. 447–468.
[27] K. Kreutz-Delgado, *The Complex Gradient Operator and the CR-Calculus*, https://arxiv.org/abs/0906.4835, 2009.
[28] J. D. Lee, Y. Sun, and M. A. Saunders, *Proximal Newton-type methods for minimizing composite functions*, SIAM J. Optim., 24 (2014), pp. 1420–1443.
[29] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed., Springer Series in Operations Research and Financial Engineering, Springer, New York, 2006.
[30] S. Paternain, A. Mokhtari, and A. Ribeiro, *A Second Order Method for Nonconvex Optimization*, preprint, arXiv:1707.08028, 2017.
[31] C. Qi, *Numerical optimization methods on Riemannian manifolds*, Ph.D. dissertation, Florida State University, Tallahassee, 2011.
[32] W. Ring and B. Wirth, *Optimization methods on Riemannian manifolds and their application to shape space*, SIAM J. Optim., 22 (2012), pp. 596–627.
[33] S. T. Smith, *Optimization techniques on Riemannian manifolds*, Fields Inst. Commun., 3 (1994), pp. 113–136.
[34] T. Steihaug, *The conjugate gradient method and trust regions in large scale optimization*, SIAM J. Numer. Anal., 20 (1983), pp. 626–637.
[35] C. Udriste, *Convex Functions and Optimization Methods on Riemannian Manifolds*, Vol. 297, Springer Science & Business Media, New York, 1994.
[36] M. Ulbrich, Z. Wen, C. Yang, D. Klöckner, and Z. Lu, *A proximal gradient method for ensemble density functional theory*, SIAM J. Sci. Comput., 37 (2015), pp. A1975–A2002.
[37] B. Vandereycken, *Low-rank matrix completion by Riemannian optimization*, SIAM J. Optim., 23 (2013), pp. 1214–1236.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[38] Z. Wen, A. Milzarek, M. Ulbrich, and H. Zhang, *Adaptive regularized self-consistent field iteration with exact Hessian for electronic structure calculation*, SIAM J. Sci. Comput., 35 (2013), pp. A1299–A1324.

[39] Z. Wen and W. Yin, *A feasible method for optimization with orthogonality constraints*, Math. Program., 142 (2013), pp. 397–434.

[40] X. Wu, Z. Wen, and W. Bao, *A regularized Newton method for computing ground states of Bose-Einstein condensates*, J. Sci. Comput., 73 (2017), pp. 303–329.

[41] C. Yang, J. C. Meza, B. Lee, and L.-W. Wang, KSSOLV—*A MATLAB toolbox for solving the Kohn-Sham equations*, ACM Trans. Math. Softw., 36 (2009), pp. 1–35.

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

[43] Y. Yang, *Globally convergent optimization algorithms on Riemannian manifolds: Uniform framework for unconstrained and constrained optimization*, J. Optim. Theory Appl., 132 (2007), pp. 245–265.

[44] H. Zhang and W. W. Hager, *A nonmonotone line search technique and its application to unconstrained optimization*, SIAM J. Optim., 14 (2004), pp. 1043–1056.

[45] X. Zhang, J. Zhu, Z. Wen, and A. Zhou, *Gradient type optimization methods for electronic structure calculations*, SIAM J. Sci. Comput., 36 (2014), pp. C265–C289.

Downloaded 04/19/20 to 124.16.148.9. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.