Blind Deconvolution by a Steepest Descent Algorithm on a Quotient Manifold

Wen Huang * Paul Hand [†]

April 17, 2018

Abstract

In this paper, we propose a Riemannian steepest descent method for solving a blind deconvolution problem. We prove that the proposed algorithm with an appropriate initialization will recover the exact solution with high probability when the number of measurements is, up to log-factors, the information-theoretical minimum scaling. The quotient structure in our formulation yields a simpler penalty term in the cost function compared to [LLSW16], which eases the convergence analysis and yields a natural implementation. Empirically, the proposed algorithm has better performance than the Wirtinger gradient descent algorithm and an alternating minimization algorithm in the sense that i) it needs fewer operations, such as DFTs and matrix-vector multiplications, to reach a similar accuracy, and ii) it has a higher probability of successful recovery in synthetic tests. An image deblurring problem is also used to demonstrate the efficiency and effectiveness of the proposed algorithm.

1 Introduction

We consider the problem of separating two unknown signals $\mathbf{w} \in \mathbb{C}^L$ and $\mathbf{x} \in \mathbb{C}^L$ given their circular convolution $\mathbf{y} \in \mathbb{C}^L$, which is an instance of blind deconvolution. Blind deconvolution is of interest due to many applications, e.g., astronomy, medical imaging, optics and communications engineering [JC93, WP98, CE07, LWDF11, WBSJ15]. The blind deconvolution problem is ill-posed without any additional assumptions. One commonly-used assumption is to suppose that the two signals \mathbf{w} and \mathbf{x} belong to known subspaces [ARR14, LLJB15, LS15]. Specifically, the signal \mathbf{w} and \mathbf{x} can be written as

$$\mathbf{w} = \mathbf{B}h, h \in \mathbb{C}^K \text{ and } \mathbf{x} = \overline{\mathbf{C}m}, m \in \mathbb{C}^N,$$

for some matrices $\mathbf{B} \in \mathbb{C}^{L \times K}$ and $\mathbf{C} \in \mathbb{C}^{L \times N}$, where overbar denotes the complex conjugate. Therefore, the blind deconvolution model with noise $\mathbf{e} \in \mathbb{C}^{L}$ is to find h and m such that

$$\mathbf{y} = \mathbf{w} * \mathbf{x} + \mathbf{e} = \mathbf{B}h * \mathbf{C}m + \mathbf{e}, \tag{1.1}$$

given $\mathbf{y} \in \mathbb{C}^L$, $\mathbf{B} \in \mathbb{C}^{L \times K}$ and $\mathbf{C} \in \mathbb{C}^{L \times N}$, where * denotes circular convolution, i.e., $\mathbf{y}_i = \sum_{j=1}^{L} \mathbf{w}_j \mathbf{x}_{i+1-j}$ and the subscript i+1-j of \mathbf{x} is understood to be modulo $\{1, 2, \ldots, L\}$. For theoretical and numerical purposes, we express (1.1) in the Fourier domain, see e.g., [ARR14, LLSW16].

^{*}huwst
08@gmail.com, Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005-1827, USA

[†]Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005-1827, USA

Let **F** denote the $L \times L$ unitary Discrete Fourier Transform (DFT) matrix. Taking Fourier transform for both sides of (1.1) yields

$$\frac{\mathbf{F}\mathbf{y}}{\sqrt{L}} = (\mathbf{F}\mathbf{B}h) \odot (\mathbf{F}\overline{\mathbf{C}m}) + \frac{\mathbf{F}\mathbf{e}}{\sqrt{L}},\tag{1.2}$$

where \odot denotes the Hadamard product. Throughout this paper, we denote \mathbf{Fy}/\sqrt{L} , \mathbf{FB} , \mathbf{FC} , and \mathbf{Fe}/\sqrt{L} by y, B, \overline{C} , and e respectively. Therefore, (1.2) becomes $y = Bh \odot \overline{Cm} + e$ and the blind deconvolution problem is equivalent to

find
$$h \in \mathbb{C}^K, m \in \mathbb{C}^N$$
 such that $y = Bh \odot \overline{Cm} + e.$ (1.3)

In recent years, two important frameworks [ARR14, LLSW16] have been proposed to solve (1.3) and to admit a rigorous recovery guarantee. In [ARR14], a convex optimization framework is proposed for the blind deconvolution problem. Specifically, the problem (1.3) is reformulated into a problem of recovering a rank-one matrix from an underdetermined system of linear equations by the well-known lifting trick¹. Replacing the rank-one constraint with a nuclear norm penalty converts the original problem into a semidefinite program. Theoretical results given in the paper show that this program enjoys a recovery guarantee under reasonable condition, such as appropriate sample complexity. The primary drawback is that the computational cost of the semidefinite program is high for large-scale problems since the lifting trick squares the dimension of the problem. The paper indeed proposes an algorithm which is based on the matrix-factorization in [BM03]; however, the convergence guarantee to the solution is not provided therein. The paper [LLSW16] overcomes the difficulty by resorting to a nonconvex approach directly. The idea closely follows from [CLS16], which shows that with high probability i) a point sufficiently close to the solution can be computed, and ii) the proposed algorithm converges to the desired solution with the computed point as the initial iterate. The cost function in [LLSW16] is

$$F(h,m) = \|y - Bh \odot \overline{Cm}\|_2^2 = \|y - \operatorname{diag}(Bhm^*C^*)\|_2^2.$$
(1.4)

The minimizer of F(h,m) is not unique since a solution (h_{\sharp}, m_{\sharp}) implies that $(h_{\sharp}p^{-1}, m_{\sharp}p)$, for any $p \in \mathbb{C}, p \neq 0$ is also a solution. To ensure numerical stability, the authors in [LLSW16] add a penalty function to (1.4) so that $||h_{\sharp}||_2$ and $||m_{\sharp}||_2$ are roughly equal in magnitude.² Note that $hm^*, h \neq 0, m \neq 0$ is a rank-one matrix. The cost function (1.4) is well-defined as a function:

$$f: \mathbb{C}_1^{K \times N} \to \mathbb{R}$$
$$X \mapsto \|y - \operatorname{diag}(BXC^*)\|_2^2.$$

where $\mathbb{C}_{1}^{K \times N} = \{X \in \mathbb{C}^{K \times N} \mid \operatorname{rank}(X) = 1\}$ denotes the set of rank-one matrices. In practice, working $\mathbb{C}^{K \times N}$ is usually inefficient especially when K and N are large. Therefore, we resort to the quotient manifold $\mathbb{C}_{*}^{K} \times \mathbb{C}_{*}^{N}/\mathbb{C}_{*}$ since this manifold is a diffeomorphism to $\mathbb{C}_{1}^{K \times N}$, where \mathbb{C}_{*}^{n} denotes a non-zero n-dimensional vector (the subscript n is dropped when n = 1), and the definition of $\mathbb{C}_{*}^{K} \times \mathbb{C}_{*}^{N}/\mathbb{C}_{*}$ is given in Sections 2 and 4.1. A representation of a point in $\mathbb{C}_{*}^{K} \times \mathbb{C}_{*}^{N}/\mathbb{C}_{*}$ is in the space $\mathbb{C}_{*}^{K} \times \mathbb{C}_{*}^{N}$, which requires much smaller storage than $\mathbb{C}^{K \times N}$. By slightly abusing notation,

¹In [ARR14], the vectors h and m are in real domain, i.e., $h \in \mathbb{R}^{K}$ and $m \in \mathbb{R}^{N}$.

²The penalty term in [LLSW16] also includes a term for coherence between h and the rows of B. This will be discussed later.

the cost function defined on the quotient manifold is

$$f: \mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* \to \mathbb{R}$$

$$\pi(h, m) \mapsto \|y - \operatorname{diag}(Bhm^* C^*)\|_2^2, \qquad (1.5)$$

where $\pi(h, m)$ denotes the point in the quotient manifold corresponding to (h, m). We refer to Sections 2 and 4.1 for the more detailed discussions about the quotient manifold. One can apply manifold optimization techniques to find the minimizer of (1.5) directly.

In this paper, we propose a Riemannian steepest descent method for solving the blind deconvolution problem (1.1). We prove that the proposed algorithm with an appropriate initialization will recover the exact solution with high probability when the number of measurements is, up to log-factors, the information-theoretical minimum scaling. The quotient structure in our formulation yields a simpler penalty term in the cost function compared to [LLSW16], which eases the convergence analysis and yields a natural implementation. In addition, unlike the cost function in [LLSW16], the objective f is strongly convex in a neighborhood of the solution, which yields efficiency in optimization algorithms since the convergence rate of a steepest descent method is related to the condition number of the Hessian at the minimizer [NW06, AMS08]. Empirically, the proposed algorithm has better performance than the Wirtinger gradient descent algorithm in [LLSW16] and an alternating minimization algorithm, see e.g., [YK94] in the sense that i) it needs fewer operations, such as DFTs and matrix-vector multiplications, to reach a similar accuracy, and ii) it has a higher probability of successful recovery.

Applying Riemannian optimization methods for solving signal processing problems, of course, is not new. To name a few, a Riemannian trust-region Newton method is used in [BSA13] to solve the synchronization of rotations problem. Various Riemannian methods are used to solve the matrix completion problem [BA11, Van13, Mis14, HAGH16, WCCL16]. A limited memory version of Riemannian BFGS method with adaptive rank strategy is used to solve the phase retrieval problem [HGZ17]. It has been shown that Riemannian methods perform very well and are competitive, and sometimes the best, methods in many applications.

In this paper, we will use the following notation. $\mathbb{C}_r^{K\times N} = \{X \in \mathbb{C}^{K\times N} \mid \operatorname{rank}(X) = r\}$ denotes the set of fixed rank matrices. $\mathbb{C}_*^{n\times p} = \{X \in \mathbb{C}^{n\times p} \mid \operatorname{rank}(X) = p\}$ denotes the noncompact Stiefel manifold. In particular, \mathbb{C}_*^n denotes the set of non-zero vectors. $\operatorname{GL}(r) = \{X \in \mathbb{C}^{r\times r} \mid \operatorname{rank}(X) = r\}$ denotes the generalized linear group. Superscript * denotes the conjugate transpose operator. Overbar $\overline{\cdot}$ denotes the complex conjugate. B is denoted by $[b_1b_2 \dots b_L]^*$ and $[b_1b_2 \dots b_K]$; C denotes $[c_1c_2 \dots c_L]^*$ and $[\mathfrak{c}_1\mathfrak{c}_2 \dots \mathfrak{c}_N]$; \mathcal{A} denote the linear operator $\mathcal{A}(Z) = \{b_l^*Zc_l\}_{l=1}^L = \operatorname{diag}(BZC^*)$. The adjoint operator \mathcal{A}^* therefore is $\mathcal{A}^*(z) = \sum_{l=1}^L z_l b_l c_l^* = B^* \operatorname{diag}(z)C$. Given a function $\mathbf{f} : \mathcal{M} \to \mathbb{R}$ where $\mathcal{M} \subset \mathbb{C}^n$ is a Riemannian manifold, $\nabla \mathbf{f}$ denotes the Euclidean gradient of \mathbf{f} , i.e., with respect to the metric $\langle \eta, \xi \rangle_2 = \operatorname{Re}(\operatorname{trace}(\eta^*\xi))$ and grad \mathbf{f} denotes the Riemannian gradient of \mathbf{f} , i.e., with respect to the Riemannian metric of \mathcal{M} . Given a function $\mathbf{h} : \mathcal{M}_1 \to \mathcal{M}_2$, notation $\operatorname{D} \mathbf{h}(x)[\eta_x]$ denotes the directional derivative of h at x along direction $\eta_x \in \operatorname{T}_x \mathcal{M}_1$, where \mathcal{M}_1 and \mathcal{M}_2 are two manifolds, and $\operatorname{T}_x \mathcal{M}_1$ is the tangent space of \mathcal{M}_1 at x.

This paper is organized as follows. Section 2 presents the Riemannian framework. Section 3 gives the initialization, proposed Riemannian method and its convergent results. Section 4 derives the Riemannian ingredients of the quotient manifold. Section 5 reports numerical experiments and Section 6 states the conclusion.

2 Problem Statement

In order to state the method, we first introduce coherence. As observed in [ARR14, LLSW16], the number of measurements required for solving the blind deconvolution problem depends on how much h_{\sharp} is correlated with the rows of the matrix B. This phenomenon also holds for the Riemannian framework. The coherence between the rows of B and h_{\sharp} is defined as³

$$\mu_h^2 = \frac{L \|Bh_{\sharp}\|_{\infty}^2}{\|h_{\sharp}\|_2^2}$$

Note that a small coherence μ_h is preferred.

For a fixed-rank manifold in real space, multiple representations have been proposed to represent a point in $\mathbb{R}_r^{K \times N}$, see [Van13, Mis14]. The same idea can be applied for the fixed-rank manifold in complex space. In this paper, the quotient manifold $\mathbb{C}_*^K \times \mathbb{C}_*^N / \operatorname{GL}(1) = \mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_*$ is used, where $\mathbb{C}_*^K \times \mathbb{C}_*^N / \mathbb{C}_* = \{[(h,m)] \mid (h,m) \in \mathbb{C}_*^K \times \mathbb{C}_*^N\}$, and $[(h,m)] = \{(hp^{-1},mp^*) \mid p \in \mathbb{C}_*\}$ is the orbit of an element $(h,m) \in \mathbb{C}_*^K \times \mathbb{C}_*^N$ under the the group \mathbb{C}_* . The group action of \mathbb{C}_* on $\mathbb{C}_*^K \times \mathbb{C}_*^N$ is $(h,m) \bullet p = (hp^{-1},mp^*)$, where $p \in \mathbb{C}_*$ and $(h,m) \in \mathbb{C}_*^K \times \mathbb{C}_*^N$. Note that the manifolds here are over \mathbb{R} , i.e., it has a real parameterization.

The cost function that we consider in the Riemannian framework is

$$\tilde{f} : \mathbb{C}^{K}_{*} \times \mathbb{C}^{N}_{*} / \mathbb{C}_{*} \to \mathbb{R}$$

$$\pi(h, m) \mapsto \|y - \operatorname{diag}(Bhm^{*}C^{*})\|_{2}^{2} + G(\pi(h, m)), \qquad (2.1)$$

where

$$G(\pi(h,m)) = \rho \sum_{i=1}^{L} G_0\left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2}\right),$$
(2.2)

 $G_0(t) = \max(t-1,0)^2$ is a C^1 function, $\rho \ge d^2 + 2.5 \|e\|_2^2$, $\frac{9}{10}d_* \le d \le \frac{11}{10}d_*$, $\mu \ge \mu_h := \sqrt{L}\|Bh_{\sharp}\|_{\infty}/\|h_{\sharp}\|_2$ and $d_* = \|h_{\sharp}m_{\sharp}^T\|_F = \|h_{\sharp}\|_2\|m_{\sharp}\|_2$. The penalty term G promotes a small coherence. The parameter d can be computed by a spectral initialization (see Algorithm 1). The value of ρ can be selected by $\rho \ge d^2 + 2.5 \|e\|_2^2$ and in the case that e is Gaussian, we choose $\rho \approx d^2 + 2.5\sigma^2 d_*^2$ where $\|e\|_2^2 \sim \frac{\sigma^2 d_*^2}{2L}\chi_{2L}^2$, χ_{2L}^2 is the χ -squared distribution and therefore $\|e\|_2^2$ concentrates around $\sigma^2 d_*^2$. The parameter μ can be selected as described in the last paragraph of [LLSW16, Section 3.2].

The framework (2.1)- (2.2) is a simplified version of that in [LLSW16], where the cost function is

$$\tilde{F} : \mathbb{C}^K \times \mathbb{C}^N \to \mathbb{R}$$

$$(h,m) \mapsto \tilde{F}(h,x) = \|y - \operatorname{diag}(Bhm^*C^*)\|_2^2 + \tilde{G}(h,m), \qquad (2.3)$$

where

$$\tilde{G}(h,m) = \rho \left[G_0 \left(\frac{\|h\|_2^2}{2d} \right) + G_0 \left(\frac{\|m\|_2^2}{2d} \right) + \sum_{i=1}^L G_0 \left(\frac{L|b_i^*h|^2}{8d\mu^2} \right) \right]$$
(2.4)

is the penalty term, G_0 , ρ , d, μ are defined as those in (2.2). The first two terms in \tilde{G} penalize large values of $||h||_2$ and $||m||_2$. It follows that the norms of h and m are not different too much,

³In [LLSW16, (3.2)], the authors use the term 'incoherence' rather than 'coherence'.

which avoids a numerical stability problem. The third term in \tilde{G} comes from an earlier work [SL16] for matrix completion. Note that a discussion about the choices of G_0 , ρ , d and μ has been given in [LLSW16, Sections 3.1 and 3.2].

The framework in [LLSW16] and the Riemannian framework are similar in the sense that their cost functions, (2.1) and (2.3), are non-convex. Here, we emphasize their differences:

- the penalty term in (2.1) scales by $||m||_2^2$ so that it is well-defined on the quotient manifold.
- the penalty term in (2.1) does not directly penalize $||m||_2$ and $||h||_2$ because the iterates $\{\pi(h_k, m_k)\}$ generated by an optimization algorithm is invariant to the representation in orbits $\{[h_k, m_k]\}$. Therefore, one can choose any representation, such as (h_k, m_k) satisfying $||h_k||_2 = ||m_k||_2$.

3 Optimization Algorithm and Theoretical Results

In this section, we state the proposed optimization algorithm for minimizing the function (2.1) as well as the main theoretical results. Because the cost function (2.1) is nonconvex, the proposed algorithm has two phases: initialization and Riemannian steepest descent method stated in Sections 3.1 and 3.2 respectively.

We consider the following model:

- the noise e is drawn from the Gaussian distribution $\mathbb{N}(0, \frac{\sigma^2 d_*^2}{2L}I_L) + \sqrt{-1}\mathbb{N}(0, \frac{\sigma^2 d_*^2}{2L}I_L);$
- C is a complex Gaussian distribution, i.e., $C_{ij} \sim \mathbb{N}(0, \frac{1}{2}) + \sqrt{-1}\mathbb{N}(0, \frac{1}{2})$; and
- B satisfies $B^*B = I_K$ and $||b_i||_2^2 \le \phi_L^K$, $i = 1, \ldots, L$ for some constant ϕ .

The Riemannian steepest descent method that we use requires an initial iterate sufficiently close to the desired solution. Two neighborhoods of the desired solution in the quotient manifold $\mathbb{C}_*^K \times \mathbb{C}_*^N/\mathbb{C}_*$ are defined as

$$\Omega_{\mu} = \{\pi(h,m) \mid \sqrt{L} \|Bh\|_{\infty} \|m\|_{2} \le 4d_{*}\mu\} \text{ and } \Pi_{\varepsilon} = \{\pi(h,m) \mid \|hm^{*} - h_{\sharp}m_{\sharp}^{*}\|_{F} \le \varepsilon d_{*}\},$$

where d_* and μ are defined in (2.3) and ε is assumed to be smaller than 1/15 throughout this paper.

3.1 Initialization

We consider the initialization by the spectral method in [LLSW16]. The following algorithm and theorem have been given therein and we give them here in Algorithm 1 and Theorem 3.1 for completeness.

Algorithm 1 Initialization via spectral method and projection

Output: (h_0, m_0) and d

1: Compute $\mathcal{A}^*(y)$;

2: Find the leading singular value d, left singular vector h_0 and right singular vector \tilde{m}_0 of $\mathcal{A}^*(y)$;

3: Find $h_0 = \arg \min_z \|z - \sqrt{d\tilde{h}_0}\|_2^2$ such that $\sqrt{L} \|Bz\|_{\infty} \leq 2\sqrt{d\mu}$ and set $m_0 = \sqrt{d\tilde{m}_0}$;

Theorem 3.1 (Initialization [LLSW16]). The initialization obtained via Algorithm 1 satisfies⁴

$$\pi(h_0, m_0) \in \Omega_{\frac{1}{2}\mu} \cap \Pi_{\frac{2}{\epsilon}\varepsilon}$$
(3.1)

and

$$\frac{9}{10}d_* \le d \le \frac{11}{10}d_*$$

holds with probability at least $1 - L^{-\gamma}$ if the number of measurements satisfies

$$L \ge C_{\gamma}(\mu_h^2 + \sigma^2) \max(K, N) \log^2(L) / \varepsilon,$$

where ε is any predetermined constant in $(0, \frac{1}{15}]$, and C_{γ} is a constant only linearly depending on γ with $\gamma \geq 1$.

3.2 Convergence Analysis

The proposed algorithm is stated in Algorithm 2, which is an implementation of a Riemannian steepest descent method. Theorem 3.2 states that if the initial iterate is close enough to the desired solution and has sufficiently low coherence, then with high probability, the sequence $\{\pi(h_k, m_k)\}$ generated by Algorithm 2 converges to the desired solution linearly, up to noise. The proof of Theorem 3.2 is given in Appendices A and B.

Algorithm	2	An	imp	lement	tation	of	аI	Riema	nnian	steep	\mathbf{est}	descent	method	l

- 1: Given h_0, m_0 , and set $k \leftarrow 0$; 2: **for** k = 0, 1, 2, ... **do**
- 3: Set

$$d_k = \|h_k\|_2 \|m_k\|_2, \ h_k \leftarrow \sqrt{d_k} \frac{h_k}{\|h_k\|_2}; \ m_k \leftarrow \sqrt{d_k} \frac{m_k}{\|m_2\|_2}$$
(3.2)

and

$$(h_{k+1}, m_{k+1}) = (h_k, m_k) - \frac{\alpha}{d_k} \left(\nabla_{h_k} \tilde{f}(h_k, m_k), \nabla_{m_k} \tilde{f}(h_k, m_k) \right)$$
(3.3)

where $\nabla_h \tilde{f}(h,m) = J_1 m + \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^{L} G'_0 \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) (b_i b_i^* h ||m||_2^2)$ is the Euclidean gradient with respect to h, $\nabla_m \tilde{f}(h,m) = J_1^* h + \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^{L} G'_0 \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) (m|b_i^*h|^2)$ is the Euclidean gradient with respect to m, and $J_1 = 2 (B^* \operatorname{diag}(\operatorname{diag}(Bhm^*C^*) - y)C)$. 4: end for

Theorem 3.2. Suppose the initialization $\pi(h_0, m_0) \in \Omega_{\frac{1}{2}\mu} \cap \prod_{\frac{2}{5}\varepsilon} and L \geq C_{\gamma}(\mu^2 + \sigma^2) \max(K, N) \log^2(L)/\varepsilon^2$. Then the iterates generated by Algorithm 2 convergence linearly to $\pi(h_{\sharp}, m_{\sharp})$ in the sense that with probability at least $1 - 4L^{-\gamma} - \frac{1}{\gamma} \exp(-(K+N))$, it holds that

$$\|h_k m_k^* - h_{\sharp} m_{\sharp}^*\|_F \le \frac{2}{3} \left(1 - \frac{\alpha a_0}{2}\right)^{k/2} \varepsilon d_* + 50 \|\mathcal{A}^*(e)\|_2,$$

⁴It follows from [LLSW16, Theorem 3.1] that one representation in the orbit $[(h_0, m_0)]$ is in the neighborhood $\Omega_{\frac{1}{2}\mu} \cap \Pi_{\frac{2}{5}\varepsilon}$. Since $\Omega_{\frac{1}{2}\mu} \cap \Pi_{\frac{2}{5}\varepsilon}$ is independent of representation, (3.1) holds.

where $a_0 = 1/1500$, $\alpha < 1/(2a_L)$ is a fixed step size, a_L is the smoothness constant of the Riemannian objective as defined by Condition A.4,

$$\|\mathcal{A}^*(e)\|_2 \le \psi \sigma d_* \max\left(\sqrt{\frac{(\gamma+1)\max(K,N)\log(L)}{L}}, \frac{(\gamma+1)\sqrt{KN}\log^2(L)}{L}\right),$$

and $\psi > 0$ is a constant.

The convergence rate of Algorithm 2 is determined by the step size α . If $\mu^2 = O(\frac{L}{(K+N)\log^2 L})$ and $\rho \approx d^2 + 2.5 ||e||_2^2$, then $a_L = O((1 + \sigma^2)(K + N)\log^2 L)$. See more details in Section B and [LLSW16, Page 13].

We emphasize the differences between Algorithm 2 and the algorithm in [LLSW16]. The steepest descent method in [LLSW16] is stated in Algorithm 3, where $(\nabla_{h_k}^w \tilde{F}, \nabla_{m_k}^w \tilde{F})$ denotes the Wirtinger derivatives. It can be shown that the Wirtinger derivatives $\left(\nabla_{h_k}^w \tilde{F}(h_k, m_k), \nabla_{m_k}^w \tilde{F}(h_k, m_k)\right)$ and Euclidean gradient $\left(\nabla_{h_k}\tilde{F}(h_k, m_k), \nabla_{m_k}\tilde{F}(h_k, m_k)\right)$ satisfy

$$\left(\nabla_{h_k}^w \tilde{F}(h_k, m_k), \nabla_{m_k}^w \tilde{F}(h_k, m_k)\right) = \frac{1}{2} \left(\nabla_{h_k} \tilde{F}(h_k, m_k), \nabla_{m_k} \tilde{F}(h_k, m_k)\right),$$
(3.4)

which yields (3.5). The differences are given as follows:

- 1. The gradient of \tilde{f} in Algorithm 2 is different from the gradient of \tilde{F} in Algorithm 3 due to the differences in their penalty terms;
- 2. The coefficient of the Euclidean gradient in Algorithm 2 is related to the norms of h_k and m_k while the coefficient in Algorithm 3 is a constant;
- 3. The h_k and m_k are normalized in Algorithm 2 so that h_k and m_k have the same norm while the norms of h_k and m_k can be different in Algorithm 3.

Algorithm 3 The steepest descent method in [LLSW16, Algorithm 2]

1: $k \leftarrow 0$; 2: **for** k = 0, 1, 2, ... **do** 3: Set

$$(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha (\nabla_{h_k}^w \tilde{F}(h_k, m_k), \nabla_{m_k}^w \tilde{F}(h_k, m_k))$$

= $(h_k, m_k) - \frac{\alpha}{2} (\nabla_{h_k} \tilde{F}(h_k, m_k), \nabla_{m_k} \tilde{F}(h_k, m_k)).$ (3.5)

4: end for

In order to understand the behavior of a generic Riemannian optimization algorithm, such as Newton method, quasi-Newton method, CG method etc, in a small neighborhood of the solution, we give the following remark, which will be formally stated and proven in Theorem 4.1 after we introduce appropriate Riemannian tools. **Remark 3.1.** With high probability provided L large enough, i) a Riemannian version of the Hessian of f in (1.5) at $\pi(h_{\sharp}, m_{\sharp})$ is positive definite, ii) there exists a lower bound, not dependent on K and N, of the smallest eigenvalue of the Riemannian Hessian, and iii) the condition number of the Riemannian Hessian at $\pi(h_{\sharp}, m_{\sharp})$ is small.

The results in Remark 3.1 have important implications. Positive definiteness of the Hessian at $\pi(h_{\sharp}, m_{\sharp})$ implies that there exists a neighborhood of $\pi(h_{\sharp}, m_{\sharp})$ in which f has a unique minimizer. Therefore, if an optimization algorithm generates iterates that converge and stay in the neighborhood, then the limit must be $\pi(h_{\sharp}, m_{\sharp})$. The independence of the lower bound of the smallest eigenvalue from K and N indicates that it is worthwhile investigating whether the size of the neighborhood depends on K and N. Note that it is unknown if the neighborhood is in the basin of attraction $\Omega_{\frac{1}{2}\mu} \cap \Pi_{\frac{2}{5}\varepsilon}$ in Theorem 3.2. We leave this work for future research. The good condition number of the Hessian at the minimizer implies that a Riemannian steepest descent method can converge quickly.

4 Riemannian Optimization

Recently many Riemannian optimization methods have been systemically analyzed and efficient libraries have been designed, e.g., Riemannian trust-region Newton method (RTR-Newton) [Bak08], Riemannian Broyden family method including BFGS method and its limited-memory version (RBroyden family, RBFGS, LRBFGS) [RW12, Hua13, HGA15, HAG18], Riemannian trust-region symmetric rank-one update method and its limited-memory version (RTR-SR1, LRTR-SR1) [Hua13, HAG15], Riemannian Newton method (RNewton) and Riemannian non-linear conjugate gradient method (RCG) [AMS08, SI15, Sat16, Zhu17].

This section follows the optimization framework on quotient manifolds in [AMS08] and uses it to derive the tools for optimization problems on $\mathbb{C}^K_* \times \mathbb{C}^N_* / \mathbb{C}_*$. Specifically, we describe the representation of the manifold $\mathbb{C}^{K \times N}_r$ for $r \geq 1$, and the ingredients which are required in Riemannian optimization, such as tangent space, retraction, vector transport and gradient. We use only r = 1in this paper for the blind deconvolution problem. The results for r > 1 are also provided since deriving them does not require much more work and is interesting for future research.⁵

4.1 Fixed-rank Manifold and Two-Factor Representation

A point on a quotient manifold is an equivalence class, which is often cumbersome computationally. In practice, choosing a representative for an equivalence class and definitions of related mathematical objects have been developed in many papers in the literature of computation on manifolds, e.g., [AMS08].

Let \mathcal{N}_r denote the total space $\mathbb{C}_*^{K \times r} \times \mathbb{C}_*^{N \times r}$ and \mathcal{Q}_r denote the quotient space $\mathcal{N}_r/\operatorname{GL}(r)$. Define a function $\pi : \mathcal{N}_r \to \mathcal{Q}_r : (H, M) \mapsto \pi(H, M)$. Let $\pi(H, M)$ denote $[(H, M)] = \{(HP^{-1}, MP^*) \mid P \in \operatorname{GL}(r)\}$ viewed as an element of \mathcal{Q}_r and $\pi^{-1}(\pi(H, M))$ is used to denote [(H, M)] viewed as a subset of \mathcal{N}_r . Let $\operatorname{T}_{(H,M)} \mathcal{N}_r$ denote the tangent space of \mathcal{N}_r at (H, M). Given $\zeta_{(H,M)} \in \operatorname{T}_{(H,M)} \mathcal{N}_r$, denote the first and second components of $\zeta_{(H,M)}$ by ζ_H and ζ_M , i.e., $\zeta_{(H,M)} = (\zeta_H, \zeta_M)$, where

⁵The following provides an example of a problem where taking r > 1 is useful. As shown in [HGZ17], it is known that the desired solution for the phase retrieval problem by a lifting approach is a rank-1 matrix. If the number of measurements is small, then optimizing over a fixed-rank manifold with rank greater than 1 is empirically shown to be more efficient and more effective than r = 1.

 $\zeta_H \in \mathbb{C}^{K \times r}$ and $\zeta_M \in \mathbb{C}^{N \times r}$. The vertical space at a point $(H, M) \in \pi^{-1}(\pi(H, M))$ is defined to be the tangent space of $\pi^{-1}(\pi(H, M))$ at (H, M). That is

$$\mathcal{V}_{(H,M)} = \{ (-H\Lambda, M\Lambda^*) \mid \Lambda \in \mathbb{C}^{r \times r} \}.$$

The Riemannian metric of \mathcal{N}_r that we use is

$$g_{(H,M)}\Big(\eta_{(H,M)},\xi_{(H,M)}\Big) = \operatorname{Re}\left(\operatorname{trace}\left(\eta_{H}^{*}\xi_{H}(M^{*}M) + \eta_{M}^{*}\xi_{M}(H^{*}H)\right)\right),\tag{4.1}$$

where $\eta_{(H,M)}, \xi_{(H,M)} \in T_{(H,M)} \mathcal{N}_r$. It follows that the induced norm is

$$\|\eta_{(H,M)}\|_{g_{(H,M)}} = \sqrt{g_{(H,M)}\left(\eta_{(H,M)},\eta_{(H,M)}\right)}.$$

We remove the subscript (H, M) of g in $\|\eta_{(H,M)}\|_{g_{(H,M)}}$, i.e., use $\|\eta_{(H,M)}\|_g$, when there is no confusion.

The idea of using the Riemannian metric (4.1) is not new. A similar metric on the real fixed-rank manifold has been used in [Mis14]. The Euclidean metric $\langle \eta_{(H,M)}, \xi_{(H,M)} \rangle_2 = \text{Re} (\text{trace} (\eta_H^* \xi_H + \eta_M^* \xi_M))$ on \mathcal{N}_r is not used here since it is not invariant to the representative in $\pi^{-1}(\pi(H,M))$. This can be verified by Lemma 4.1. The Riemannian metric (4.1) is not equivalent to the Euclidean metric in $\mathbb{C}^{K \times N}$, i.e.,

$$g_{(H,M)}\Big(\eta_{(H,M)},\xi_{(H,M)}\Big) \neq \operatorname{Re}\left(\operatorname{trace}\left((\eta_H M^* + H\eta_M^*)^*(\xi_H M^* + H\xi_M^*)\right)\right)$$

in general. Note that the tangent vector (η_H, η_M) of \mathcal{N}_r at (H, M) corresponds to the tangent vector $\eta_H M^* + H \eta_M^*$ of $\mathbb{C}^{K \times N}$ at $H M^*$.

The horizontal space at (H, M), denoted by $\mathcal{H}_{(H,M)}$, is defined to be the subspace of $T_{(H,M)}\mathcal{N}_r$ that is orthogonal to $\mathcal{V}_{(H,M)}$ with respect to the metric (4.1). That is

$$\mathcal{H}_{(H,M)} = \left\{ \left(\begin{bmatrix} H & H_{\perp} \end{bmatrix} \begin{bmatrix} K \\ T \end{bmatrix}, \begin{bmatrix} M & M_{\perp} \end{bmatrix} \begin{bmatrix} K^* \\ Q \end{bmatrix} \right) \mid K \in \mathbb{C}^{r \times r}, T \in \mathbb{C}^{(K-r) \times r}, Q \in \mathbb{C}^{(N-r) \times r} \right\},$$

where H_{\perp} denotes a $K \times (K - r)$ orthonormal matrix such that $H^*H_{\perp} = 0$ and likewise for M_{\perp} . The horizontal space $\mathcal{H}_{(H,M)}$ is a representation of the tangent space $T_{\pi(H,M)} \mathcal{Q}_r$. It is known that for any $\eta_{\pi(H,M)} \in T_{\pi(H,M)} \mathcal{Q}_r$, there exists a unique vector in $\mathcal{H}_{(H,M)}$, called the horizontal lift of $\eta_{\pi(H,M)}$ and denoted by $\eta_{\uparrow_{(H,M)}}$, satisfying $D \pi(H,M)[\eta_{\uparrow_{(H,M)}}] = \eta_{\pi(H,M)}$, see e.g., [AMS08]. The relationship among horizontal lifts of a tangent vector $\eta_{\pi(H,M)}$ is given in Lemma 4.1. The result follows from [Hua13, Theorem 9.3.1].

Lemma 4.1. A vector field $(\hat{\theta}, \hat{\vartheta})$ on \mathcal{N}_r is the horizontal lift of a vector field on \mathcal{Q}_r if and only if, for each $(H, M) \in \mathcal{N}_r$, we have

$$\left(\hat{\theta}_{HP^{-1}}, \hat{\vartheta}_{MP^*}\right) = \left(\hat{\theta}_H P^{-1}, \hat{\vartheta}_M P^*\right),\tag{4.2}$$

for all $P \in GL(r)$.

The Riemannian metric (4.1) on \mathcal{N}_r defines a Riemannian metric on \mathcal{Q}_r .

Lemma 4.2. The following defines a Riemannian metric on Q_r :

$$g_{\pi(H,M)}\Big(\eta_{\pi(H,M)},\xi_{\pi(H,M)}\Big) = \operatorname{Re}\left(\operatorname{trace}\left(\eta_{\uparrow_{H}}^{*}\xi_{\uparrow_{H}}(M^{*}M) + \eta_{\uparrow_{M}}^{*}\xi_{\uparrow_{M}}(H^{*}H)\right)\right).$$
(4.3)

Proof. It can be easily verified that the right hand side of (4.3) is invariant to representions in $\pi^{-1}(\pi(H, M))$.

The orthogonal projections onto the horizontal space or the vertical space are similar to those in [Mis14, Table 4.3]. We give them below for completeness.

Lemma 4.3. The orthogonal projection to the vertical space $\mathcal{V}_{(H,M)}$ is $P_{(H,M)}^v\left(\eta_{(H,M)}\right) = (-H\Lambda, M\Lambda^*)$, where $\Lambda = 0.5(\eta_M^* M(M^*M)^{-1} - (H^*H)^{-1}H^*\eta_H)$. The orthogonal projection to the horizontal space $\mathcal{H}_{(H,M)}$ is $P_{(H,M)}^h\left(\eta_{(H,M)}\right) = \eta_{(H,M)} - P_{(H,M)}^v\left(\eta_{(H,M)}\right)$.

Proof. It is easy to verify that $g\left(P_{(H,M)}^{v}\left(\eta_{(H,M)}\right),P_{(H,M)}^{h}\left(\eta_{(H,M)}\right)\right)=0$ and $P_{(H,M)}^{v}\left(\eta_{(H,M)}\right)+P_{(H,M)}^{h}\left(\eta_{(H,M)}\right)=\eta_{(H,M)}$.

4.2 Retraction and Vector transport

Retraction is used to update iterates in a Riemannian algorithm and vector transport is used to compare tangent vectors in different tangent spaces. Specifically, a retraction R is a smooth mapping from the tangent bundle $T \mathcal{M}$, which is the set of all tangent vectors to \mathcal{M} , onto a manifold \mathcal{M} such that (i) $R(0_x) = x$ for all $x \in \mathcal{M}$ (where 0_x denotes the origin of $T_x \mathcal{M}$) and (ii) $\frac{d}{dt}R(t\xi_x)|_{t=0} = \xi_x$ for all $\xi_x \in T_x \mathcal{M}$. The restriction of R to $T_x \mathcal{M}$ is denoted by R_x . A vector transport $\mathcal{T} : T \mathcal{M} \oplus T \mathcal{M} \to T \mathcal{M}, (\eta_x, \xi_x) \mapsto \mathcal{T}_{\eta_x} \xi_x$ with associated retraction R is a smooth mapping such that, for all (x, η_x) in the domain of R and all $\xi_x \in T_x \mathcal{M}$, it holds that (i) $\mathcal{T}_{\eta_x} \xi_x \in T_{R(\eta_x)} \mathcal{M}$, (ii) $\mathcal{T}_{0_x} \xi_x = \xi_x$, (iii) \mathcal{T}_{η_x} is a linear map. A retraction in the total space \mathcal{N}_r is

$$R_{(H,M)}(\eta_{(H,M)}) = (H + \eta_H, M + \eta_M).$$
(4.4)

It follows from Lemma 4.1 that $\pi(R_{(H,M)}(\eta_{(H,M)})) = \pi(R_{(HP^{-1},MP^*)}(\eta_{(HP^{-1},MP^*)}))$. Therefore, (4.4) defines a retraction in the quotient manifold Q_r , i.e.,

$$R_{\pi(H,M)}(\eta_{\pi(H,M)}) := \pi(R_{(H,M)}(\eta_{\uparrow_{(H,M)}})).$$
(4.5)

The vector transport used is the vector transport by parallelization [HAG16]:

$$\mathcal{T}_{\eta_x} = \mathcal{B}_y \mathcal{B}_x^{\dagger},\tag{4.6}$$

where $y = R_x(\eta_x)$, \mathcal{B} is a smooth tangent basis field defined on an open set of a manifold \mathcal{M} and \mathcal{B}_x^{\dagger} denotes the pseudo-inverse of \mathcal{B}_x . A smooth orthonormal basis of $\mathcal{H}_{(H,M)}$ is

$$\begin{cases} \left(\frac{1}{\sqrt{2}}HL_{H}^{-*}e_{i}e_{j}^{T}L_{M}^{-1},\frac{1}{\sqrt{2}}ML_{M}^{-*}e_{j}e_{i}^{T}L_{H}^{-1}\right), i = 1, \dots, r, j = 1, \dots, r \end{cases} \\ \bigcup \left\{ \left(\frac{1}{\sqrt{2}}HL_{H}^{-*}e_{i}e_{j}^{T}L_{M}^{-1}\sqrt{-1}, -\frac{1}{\sqrt{2}}ML_{M}^{-*}e_{j}e_{i}^{T}L_{H}^{-1}\sqrt{-1}\right), i = 1, \dots, r, j = 1, \dots, r \right\} \\ \bigcup \left\{ \left(H_{\perp}\tilde{e}_{i}e_{j}^{T}L_{M}^{-1}, 0_{N\times r}\right), i = 1, \dots, K - r, j = 1, \dots, r \right\} \\ \bigcup \left\{ \left(H_{\perp}\tilde{e}_{i}e_{j}^{T}L_{M}^{-1}\sqrt{-1}, 0_{N\times r}\right), i = 1, \dots, K - r, j = 1, \dots, r \right\} \\ \bigcup \left\{ \left(0_{K\times r}, M_{\perp}\hat{e}_{i}e_{j}^{T}L_{H}^{-1}\right), i = 1, \dots, N - r, j = 1, \dots, r \right\} \\ \bigcup \left\{ \left(0_{K\times r}, M_{\perp}\hat{e}_{i}e_{j}^{T}L_{H}^{-1}\sqrt{-1}\right), i = 1, \dots, N - r, j = 1, \dots, r \right\}, \end{cases}$$

where (e_1, \ldots, e_r) is the canonical basis of \mathbb{R}^r , $(\tilde{e}_1, \ldots, \tilde{e}_{(K-r)})$ is the canonical basis of \mathbb{R}^{K-r} , $(\hat{e}_1, \ldots, \hat{e}_{(N-r)})$ is the canonical basis of \mathbb{R}^{N-r} , and $H^*H = L_H L_H^*$ and $M^*M = L_M L_M^*$ are Cholesky decompositions. Therefore, the vector transport (4.6) with the above orthonormal basis yields locally smooth, linear mapping from $\mathcal{H}_{(H,M)}$ to $\mathcal{H}_{R_{(H,M)}(\eta_{(H,M)})}$. In the case of r = 1, one can choose h_{\perp} to be the last K - 1 columns of the Householder matrix $I - 2vv^*/||v||_2^2$, where $v = h - ||h||e_1$, and similarly for m_{\perp} . It follows that any corresponding pair of tangent vectors $\eta_{(h,m)}$ and $(\eta_{hp^{-1}}, \eta_{mp^*})$ in bases $B_{(h,m)}$ and $B_{(hp^{-1},mp^*)}$ satisfies (4.2), i.e., $(\eta_{hp^{-1}}, \eta_{mp^*}) = (\eta_h p^{-1}, \eta_m p^*)$. Therefore, the mapping

$$\left(\mathcal{T}_{\eta_{\pi(h,m)}}\xi_{\pi(h,m)}\right)_{\uparrow_{(h,m)}} := B_{(\tilde{h},\tilde{m})}B^{\dagger}_{(h,m)}$$

$$(4.7)$$

defines a vector transport on \mathcal{Q}_1 , where $(\tilde{h}, \tilde{m}) = R_{(h,m)} \left(\eta_{\uparrow_{(h,m)}} \right)$.

4.3 Riemannian Gradient

The Riemannian gradient of a function $\mathbf{f} : \mathcal{M} \to \mathbb{R} : x \mapsto \mathbf{f}(x)$ is defined to be the unique tangent vector grad $\mathbf{f} \in T_x \mathcal{M}$ satisfying

$$D\mathbf{f}(x)[\eta_x] = g(\operatorname{grad} \mathbf{f}(x), \eta_x), \ \forall \eta_x \in T_x \mathcal{M},$$

where g is the Riemannian metric of \mathcal{M} . The Riemannian gradient of (2.1) is derived in Lemma 4.4.

Lemma 4.4. Given any $\pi(h,m) \in Q_1$, the gradient of $\tilde{f}_{\pi(h,m)}$ is

$$\left(\operatorname{grad} \tilde{f}(\pi(h,m))\right)_{\uparrow(h,m)} = P_{(h,m)}^{h} \left(\nabla_{h} \tilde{f}(h,m)(m^{*}m)^{-1}, \nabla_{m} \tilde{f}(h,m)(h^{*}h)^{-1}\right),$$
(4.8)

where $\nabla_h \tilde{f}(h,m) = J_1 m + \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^{L} G'_0 \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) (b_i b_i^* h ||m||_2^2)$ is the Euclidean gradient with respect to h, $\nabla_m \tilde{f}(h,m) = J_1^* h + \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^{L} G'_0 \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) (m|b_i^*h|^2)$ is the Euclidean gradient with respect to m, and $J_1 = 2 (B^* \operatorname{diag}(\operatorname{diag}(Bhm^*C^*) - y)C)$. Moreover, if the representation (h,m) satisfies $\|h\|_2 = \|m\|_2$, then

$$\left(\operatorname{grad} \tilde{f}(\pi(h,m))\right)_{\uparrow_{(h,m)}} = \left(\nabla_h \tilde{f}(h,m)(m^*m)^{-1}, \nabla_m \tilde{f}(h,m)(h^*h)^{-1}\right).$$
(4.9)

Proof. Taking directional derivative along any direction $\eta_{\pi(h,m)} \in T_{\pi(h,m)} \mathcal{Q}_r$ yields

$$\begin{split} \mathrm{D}\,\tilde{f}(\pi(h,m))[\eta_{\pi(h,m)}] &= \mathrm{D}\left(\|y - \mathrm{diag}(Bhm^*C^*)\|_2^2 + \rho \sum_{i=1}^L G_0\left(\frac{L|b_i^*h|^2 \|m\|_2^2}{8d^2\mu^2}\right) \right) [(\eta_{\uparrow_h},\eta_{\uparrow_m})] \\ &= & 2\mathrm{Re}\,\mathrm{trace}\left((\mathrm{diag}(Bhm^*C^*) - y)^T\,\mathrm{diag}(B\eta_{\uparrow_h}m^*C^*)\right) \\ &+ & 2\mathrm{Re}\,\mathrm{trace}\left((\mathrm{diag}(Bhm^*C^*) - y)^T\,\mathrm{diag}(Bh\eta_{\uparrow_m}^*C^*)\right) \\ &+ & \mathrm{Re}\left(\frac{L\rho}{4d^2\mu^2}\sum_{i=1}^L G_0'\left(\frac{L|b_i^*h|^2 \|m\|_2^2}{8d^2\mu^2}\right)(\eta_{\uparrow_h}^*b_ib_i^*h\|m\|_2^2)\right) \\ &+ & \mathrm{Re}\left(\frac{L\rho}{4d^2\mu^2}\sum_{i=1}^L G_0'\left(\frac{L|b_i^*h|^2 \|m\|_2^2}{8d^2\mu^2}\right)(\eta_{\uparrow_m}^*m|b_i^*h|^2)\right) \\ &= & \mathrm{Re}\left(\eta_{\uparrow_h}^*\nabla_h\tilde{f}(h,m) + \eta_{\uparrow_m}^*\nabla_m\tilde{f}(h,m)\right). \end{split}$$

Combining the above equation with the definition [AMS08, (3.31)] and (4.3) yields (4.8). If $||h||_2 = ||m||_2$, then the projection is not necessary since $\left(\nabla_h \tilde{f}(h,m)(m^*m)^{-1}, \nabla_m \tilde{f}(h,m)(h^*h)^{-1}\right)$ is in $\mathcal{H}_{(h,m)}$ already.

4.4 A Riemannian Steepest Descent Algorithm

Algorithm 2 is an instance of a Riemannian steepest descent method. A Riemannian steepest descent iteration for optimizing a real-valued function \mathbf{f} on \mathcal{M} is $x_{k+1} = R_{x_k}(-\alpha \operatorname{grad} \mathbf{f}(x_k))$, where $\alpha > 0$ is a step size and R is a retraction. In the case of optimizing \tilde{f} in (2.1), the iteration is $\pi(h_{(k+1)}, m_{(k+1)}) = R_{\pi(h_k, m_k)}(\operatorname{grad} \tilde{f}(\pi(h_k, m_k)))$, which yields implementations stated in Algorithm 4. The update (4.10) is the generic Riemannian steepest descent update formula and (4.11) is for this specifical problem. Since the iteration $\pi(h_{(k+1)}, m_{(k+1)}) = R_{\pi(h_k, m_k)}(\operatorname{grad} \tilde{f}(\pi(h_k, m_k)))$ is independent of representations chosen in equivalence class, we can choose h_k and m_k such that they always have the same norm. In addition, the projection onto the horizontal space is not necessary due to (4.9). It follow that we have the update formula (4.12), which is used in Algorithm 2.

4.5 Local Convexity

In this section, we give the complete version of Remark 3.1 in Theorem 4.1. We consider the Riemannian Hessian of the function $f \circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}$, which is defined on the tangent space $T_{\pi(h_{\sharp},m_{\sharp})} Q_1$. Since a tangent space is a vector space, the Riemannian Hessian with respect to the metric (4.3) is

$$\left(\operatorname{Hess} f \circ \tilde{R}_{\pi(h,m)}(\xi_{\pi(h,m)})[\eta_{\pi(h,m)}] \right)_{\uparrow_{(h,m)}}$$

= $P^{h}_{(h,m)} \left((J_{3}(m+\xi_{\uparrow_{m}})+J_{1}\eta_{\uparrow_{m}})(m^{*}m)^{-1}, (J_{3}^{*}(h+\xi_{\uparrow_{h}})+J_{1}^{*}\eta_{\uparrow_{h}})(h^{*}h)^{-1} \right),$

where J_1 is defined in (4.8) and $J_3 = 2 \operatorname{Re} \left(B^* \operatorname{diag}(\operatorname{diag}(B\left(\eta_{\uparrow_h}(m+\eta_{\uparrow_m})^* + (h++\xi_{\uparrow_h})\eta_{\uparrow_m}^*\right)C^*))C \right).$ The complete version of Remark 3.1 is given below. 1: $k \leftarrow 0;$

2: for $k = 0, 1, 2, \dots$ do

3: Set

$$(h_{k+1}, m_{k+1}) = \tilde{R}_{(h_k, m_k)} \left(-\alpha \left(\operatorname{grad} \tilde{f}(\pi(h, m)) \right)_{\uparrow_{(h, m)}} \right);$$
(4.10)

or equivalently

$$(h_{k+1}, m_{k+1}) = (h_k, m_k) - \alpha P_{(h,m)}^h \left(\nabla_{h_k} \tilde{f}(h_k, m_k) (m_k^* m_k)^{-1}, \nabla_{m_k} \tilde{f}(h_k, m_k) (h_k^* h_k)^{-1} \right);$$
(4.11)

or equivalently $d_k = \|h_k\|_2 \|m_k\|_2$, $h_k \leftarrow \sqrt{d_k} \frac{h_k}{\|h_k\|_2}$; $m_k \leftarrow \sqrt{d_k} \frac{m_k}{\|m_2\|_2}$ and

$$(h_{k+1}, m_{k+1}) = (h_k, m_k) - \frac{\alpha}{d_k} \left(\nabla_{h_k} \tilde{f}(h_k, m_k), \nabla_{m_k} \tilde{f}(h_k, m_k) \right);$$
(4.12)

4: end for

Theorem 4.1. With probability at least $1 - L^{-\gamma}$, the eigenvalues of the Hessian of $f \circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}$ at $0_{\pi(h_{\sharp},m_{\sharp})}$ are between $9d_*^2/5$ and $22d_*^2/5$ up to noise, i.e.,

$$\frac{22d_*^2}{5} + 4\|\mathcal{A}^*(e)\|_2 d_* \ge \frac{g\left(\eta_{\pi(h_{\sharp},m_{\sharp})}, \operatorname{Hess} f \circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}(0_{\pi(h_{\sharp},m_{\sharp})})[\eta_{\pi(h_{\sharp},m_{\sharp})}]\right)}{g(\eta_{\pi(h_{\sharp},m_{\sharp})},\eta_{\pi(h_{\sharp},m_{\sharp})})} \ge \frac{9d_*^2}{5} - 4\|\mathcal{A}^*(e)\|_2 d_*,$$

for all $\eta_{\pi(h_{\sharp},m_{\sharp})} \in T_{\pi(h_{\sharp},m_{\sharp})} \mathcal{Q}_r$, where \tilde{R} is defined in (4.5).

Proof. Without loss of generality, we assume $||h_{\sharp}||_2 = ||m_{\sharp}||_2 = \sqrt{d_*}$. We have

$$g\left(\eta_{\pi(h_{\sharp},m_{\sharp})},\operatorname{Hess} f\circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}(0_{\pi(h_{\sharp},m_{\sharp})})[\eta_{\pi(h_{\sharp},m_{\sharp})}]\right) = \frac{d^{2}}{dt^{2}}f\circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}(t\eta_{\pi(h_{\sharp},m_{\sharp})})|_{t=0}$$
$$= \frac{d^{2}}{dt^{2}}||y - \operatorname{diag}(B(h_{\sharp} + t\eta_{\uparrow h_{\sharp}})(m_{\sharp} + t\eta_{\uparrow m_{\sharp}})^{*}C^{*})||_{2}^{2}|_{t=0}$$
$$= \operatorname{Re}\left(\operatorname{trace}\left(\eta_{\uparrow h_{\sharp}}^{*}(J_{2}m_{\sharp} + J_{1}\eta_{\uparrow m_{\sharp}}) + \eta_{\uparrow m_{\sharp}}^{*}(J_{2}^{*}h_{\sharp} + J_{1}^{*}\eta_{\uparrow h_{\sharp}})\right)\right),$$

where J_1 is defined in (4.8) and $J_2 = 2 \operatorname{Re} \left(B^* \operatorname{diag}(\operatorname{diag}(B\left(\eta_{\uparrow_{h_{\sharp}}}m^* + h\eta^*_{\uparrow_{m_{\sharp}}}\right)C^*))C \right)$. It follows from the definition of $\pi(h_{\sharp}, m_{\sharp})$ that $J_1 = 2\mathcal{A}^*(e)$. Thus,

$$\frac{d^2}{dt^2} f \circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}(t\eta_{\pi(h_{\sharp},m_{\sharp})})|_{t=0} = \operatorname{Re}(\operatorname{trace}(2\eta_{\uparrow_{h_{\sharp}}}J_1\eta_{\uparrow_{m_{\sharp}}} + J_2^*(\eta_{\uparrow_{h_{\sharp}}}m^* + h_{\sharp}\eta_{\uparrow_{m_{\sharp}}}^*)))$$

=4Re trace $\left(\eta_{\uparrow_{h_{\sharp}}}^*\mathcal{A}^*(e)\eta_{\uparrow_{m_{\sharp}}}\right) + 2\|\mathcal{A}(\eta_{\uparrow_{h_{\sharp}}}m^* + h_{\sharp}\eta_{\uparrow_{m_{\sharp}}}^*)\|_2^2.$ (4.13)

It holds that

$$|4\operatorname{Re}\operatorname{trace}\left(\eta_{\uparrow_{h_{\sharp}}}^{*}\mathcal{A}^{*}(e)\eta_{\uparrow_{m_{\sharp}}}\right)| \leq 4\|\mathcal{A}^{*}(e)\|_{2}\|\eta_{\uparrow_{m_{\sharp}}}\|_{2}\|\eta_{\uparrow_{h_{\sharp}}}\|_{2} \leq 4\|\mathcal{A}^{*}(e)\|_{2}\left(\|\eta_{\uparrow_{m_{\sharp}}}\|_{2}^{2}+\|\eta_{\uparrow_{h_{\sharp}}}^{2}\|_{2}\right)$$
$$\leq 4\|\mathcal{A}^{*}(e)\|_{2}d_{*}g(\eta_{\pi(h_{\sharp},m_{\sharp})},\eta_{\pi(h_{\sharp},m_{\sharp})}).$$
(4.14)

Using [LLSW16, Lemma 5.12] yields

$$\frac{11}{5} \|\eta_{\uparrow_{h_{\sharp}}} m_{\sharp}^* + h_{\sharp} \eta_{\uparrow_{m_{\sharp}}}^* \|_F^2 \ge 2 \|\mathcal{A}(\eta_{\uparrow_{h_{\sharp}}} m^* + h_{\sharp} \eta_{\uparrow_{m_{\sharp}}}^*)\|_2^2 \ge \frac{9}{5} \|\eta_{\uparrow_{h_{\sharp}}} m_{\sharp}^* + h_{\sharp} \eta_{\uparrow_{m_{\sharp}}}^* \|_F^2,$$
(4.15)

with probability at least $1 - L^{-\gamma}$ provided $L \ge C_{\gamma} \max(K, \mu_h^2 N) \log^2(L)$. Note that this probability is independent of $\eta_{\pi(h_{\sharp}, m_{\sharp})}$.

Decompose $\eta_{\uparrow_{h_{\sharp}}} = ah_{\sharp} + b\eta_{\uparrow_{h_{\perp}}}$ and $\eta_{\uparrow_{m_{\sharp}}} = a^*m_{\sharp} + \beta\eta_{\uparrow_{m_{\perp}}}$, where $h^*_{\sharp}\eta_{\uparrow_{h_{\perp}}} = 0$, $\|\eta_{\uparrow_{h_{\perp}}}\|_2 = 1$, $m^*_{\sharp}\eta_{\uparrow_{m_{\perp}}} = 0$, and $\|\eta_{\uparrow_{m_{\perp}}}\|_2 = 1$. Therefore, we have

$$\begin{aligned} \|\eta_{\uparrow_{h_{\sharp}}}m^{*} + h_{\sharp}\eta_{\uparrow_{m_{\sharp}}}^{*}\|_{F}^{2} &= \|2ah_{\sharp}m_{\sharp}^{*}\|_{F}^{2} + \|b\eta_{\uparrow_{h_{\perp}}}m_{\sharp}^{*}\|_{F}^{2} + \|\beta h_{\sharp}\eta_{\uparrow_{m_{\perp}}}^{*}\|_{F}^{2} = |2a|^{2}d_{*}^{2} + b^{2}d_{*} + \beta^{2}d_{*} \\ &\geq 2|a|^{2}d_{*}^{2} + b^{2}d_{*} + \beta^{2}d_{*} = (|a|^{2}d_{*} + b^{2} + |a|^{2}d_{*} + \beta^{2})d_{*} \\ &= d_{*}(\|\eta_{\uparrow_{h_{\sharp}}}\|_{2}^{2} + \|\eta_{\uparrow_{m_{\sharp}}}\|_{2}^{2}) = d_{*}^{2}g(\eta_{\pi(h_{\sharp},m_{\sharp})},\eta_{\pi(h_{\sharp},m_{\sharp})}). \end{aligned}$$
(4.16)

Similarly, we have

$$\|\eta_{\uparrow_{h_{\sharp}}}m^{*} + h_{\sharp}\eta_{\uparrow_{m_{\sharp}}}^{*}\|_{F}^{2} = |2a|^{2}d_{*}^{2} + b^{2}d_{*} + \beta^{2}d_{*} \le 2(|a|^{2}d_{*} + b^{2} + |a|^{2}d_{*} + \beta^{2})d_{*} \le 2d_{*}^{2}g(\eta_{\pi(h_{\sharp},m_{\sharp})},\eta_{\pi(h_{\sharp},m_{\sharp})}).$$

$$(4.17)$$

Combining (4.13), (4.14), (4.15), (4.16) and (4.17) yields

$$\frac{22d_*^2}{5} + 4\|\mathcal{A}^*(e)\|_2 d_* \ge \frac{g\left(\eta_{\pi(h_{\sharp},m_{\sharp})}, \operatorname{Hess} f \circ \tilde{R}_{\pi(h_{\sharp},m_{\sharp})}(0_{\pi(h_{\sharp},m_{\sharp})})[\eta_{\pi(h_{\sharp},m_{\sharp})}]\right)}{g(\eta_{\pi(h_{\sharp},m_{\sharp})}, \eta_{\pi(h_{\sharp},m_{\sharp})})} \ge \frac{9d_*^2}{5} - 4\|\mathcal{A}^*(e)\|_2 d_*.$$

5 Experiments

In this section, numerical simulations are used to illustrate the performance of the proposed method. Section 5.1 gives the experimental environment, synthetic problem settings, parameters and complexities. The synthetic problems are used in Sections 5.2–5.4. Specifically, Section 5.2 compares the efficiency of the proposed method to the method in [LLSW16] and an alternating minimization method; Section 5.3 presents an empirical estimation for the probability of successful recovery against the number of measurements; and Section 5.4 shows the robustness of the proposed method. In Section 5.5, an image from FLAVIA dataset [WBX⁺07], rather than a synthetic data, is used to show the performance of the proposed method in an image deblurring problem.

5.1 Environment, Step Size, Problem Setting and Complexities

The codes of Algorithm 4 are written in C++ using the library ROPTLIB [HAGH16] through its Matlab interface. All experiments are performed in Matlab R2016b on a 64bit Windows system with 3.4GHz CPU (Intel(R) Core(TM) i7-6700). The DFT is performed using the library FFTW [FJ05] with one thread. The code is available at www.math.fsu.edu/~whuang2/papers/BDSDAQM.htm.

In signal recovery problems, including the blind deconvolution problem, theoretical results usually require the step size to be a sufficiently small constant. However, in practice, a too small step size slows down algorithm significantly while a too large step size makes algorithm fail to converge. Therefore, heuristic ideas have been used. In [CLS16], the step size is given by a predetermined nonincreasing sequence. In [LLSW16], the step size is chosen by backtracking with initial step size 1/d, where d is the singular value in Algorithm 1. In this paper, the step size is given by the backtracking algorithm with BB initial step size [BB88]. Specifically, suppose the iterates generated by Algorithm 2 is $\{x_k\}$ and $\eta_k = R_{x_k}^{-1}(x_{k+1})$. We use the initial step size $g(s_k, y_k)/g(y_k, y_k)$, where $s_k = \mathcal{T}_{\eta_k}\eta_k$ and $y_k = \text{grad } f(x_{k+1}) - \mathcal{T}_{\eta_k} \text{ grad } f(x_k)$ and the vector transport \mathcal{T} is defined in (4.6). Note that this initial step size may or may not be smaller than $1/(2a_L)$, which is assumed by Theorem 3.2.

The matrix *B* is the first *K* columns of a unitary $L \times L$ DFT matrix. The matrix *A* is a Gaussian random matrix. The initial iterate, ρ , and μ are given using the same method as [LLSW16], i.e., the initial iterate is the normalized leading singular vectors of $\mathcal{A}^*(y)$, $\rho = d^2/100$ and $\mu = 6\sqrt{L/(K+N)}/\log(L)$. Unless otherwise indicated, the measurements *y* are noiseless, i.e., $y = \mathcal{A}(h_{\sharp}m_{\sharp}^*)$, where h_{\sharp} and m_{\sharp} are Gaussian random vectors. The stopping criterion requires $||y - \mathcal{A}(hm^*)||_2/||y||_2 \leq 10^{-8}$.

Since $B = \sqrt{\overline{L}} \mathbf{F} \mathbf{B}$ and $C = \sqrt{\overline{L}} \mathbf{F} \mathbf{C}$, the cost function \tilde{f} is⁶

$$\tilde{f}(\pi(h,m)) = \|y - L\operatorname{diag}\left((\mathbf{FB}h)(\bar{\mathbf{FC}}m)^*\right)\|_2^2 + \rho \sum_{i=1}^L G_0\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right).$$

If B has K nonzero entries, C is a random dense matrix and the order of computations of multiplications is to compute **B**h and then apply FFT to the resulting vector $\mathbf{F}(\mathbf{B}h)$ and likewise for $\bar{\mathbf{F}}(\mathbf{C}m)$, then the complexity of a function evaluation is 2FFT + 2LN + O(L) flops, where a flop is a float point operation [GV96, Section 1.2.4]. Using the same idea, the complexity of a gradient evaluation after evaluating the function value at the same point is 3FFT + 2LN + O(L) flops. The complexity of the vector transport (4.6) is O(K+N) using the intrinsic representation (see detailed discussions about intrinsic representation in [HAG16]).

5.2 Efficiency

Algorithm 2 is compared to the algorithm in [LLSW16] and an alternating minimization algorithm. The alternating minimization algorithm is stated in Algorithm 5, which approximately optimizes over h and m alternatively. Specifically, when one of h and m is fixed, the function F in (1.4) is quadratic and the step size in Steps 3 and 4 has a closed form which can be compute cheaply. Let

Algorithm 5 An alternating minimization algorithm

1: $k \leftarrow 0$; 2: for k = 0, 1, 2, ... do 3: $t_* = \arg\min_{t>0} F(h_k, m_k - t\nabla_{m_k}F(h_k, m_k))$ and set $m_{k+1} = m_k - t_*\nabla_{m_k}F(h_k, m_k)$; 4: $t_* = \arg\min_{t>0} F(h_k - t\nabla_{h_k}F(h_k, m_{k+1}), m_{k+1})$ and set $h_{k+1} = h_k - t_*\nabla_{h_k}F(h_k, m_{k+1})$; 5: end for

NCBT denote the algorithm in [LLSW16], NCBB denote the algorithm in [LLSW16] with modified initial step size: using BB step size rather than 1/d; AMA denote the alternating minimization algorithm; and ROBB denote the proposed Algorithm 4 with the BB initial step size. NCBT, NCBB, and AMA are implemented in Matlab. The parameters K, N are set to be 100 and 100 respectively.

 $^{{}^{6}\}tilde{F}$ can be done similarly and therefore its complexity is not discussed here.

Table 1 reports the results of an average of 100 random runs for L = 400 and 600. Since the algorithms are performed on different languages, it is unfair to compare their computational time. Therefore, the machine-independent operations are reported. It is shown that using BB step size as the initial step size improves efficiency significantly. Increasing the number of measurements reduces the difficulty of the optimization problem in the sense that the numbers of various operations decrease. ROBB method outperforms the other methods in the sense that it needs fewer number of all the operations to reach a similar accuracy.

Table 1: An average of 100 random runs. nFFT denotes the number of fast Fourier transforms (or inverst FFT). nBh and nCm denote the numbers of matrix vector multiplications **B**h and **C**m respectively. Note that nBh = nCm. RMSE denotes the relative error $\frac{\|hm^T - h_{\sharp}m_{\sharp}^T\|}{\|h_{\sharp}\|\|m_{\sharp}\|}$. The subscript k indicates a scale of 10^k. The algorithms NCBT, NCBB, AMA, and ROBB are introduced in the first paragraph of Section 5.2.

		L =	400		L = 600				
Algorithms	NCBT	NCBB	AMA	ROBB	NCBT	NCBB	AMA	ROBB	
nBh/nCm	1040	349	718	208	403	162	294	122	
nFFT	2533	865	1436	518	984	401	588	303	
RMSE	3.73_{-8}	2.24_{-8}	3.67_{-8}	2.20_{-8}	2.39_{-8}	1.48_{-8}	2.34_{-8}	1.42_{-8}	

5.3 Number of Measurements vs Success Rate

Both parameters K and N are set to be 50. L/(K+L) takes 21 values: $\{1, 1.05, 1.1, \ldots, 1.45, 1.5, 1.6, \ldots, 2.4, 2.5\}$. We consider an algorithm to successfully recover $\pi(h_{\sharp}, m_{\sharp})$ if the RMSE of the final iterate $\pi(h, m)$ is less than 10^{-2} , i.e., $\|hm^* - h_{\sharp}m_{\sharp}^*\|_F / \|h_{\sharp}m_{\sharp}^*\|_F \leq 10^{-2}$.

Figure 1 shows an empirical phase transition curves for the four algorithms: NCBT, NCBB, AMA, and ROBB. NCBT and NCBB perform similarly while AMA and ROBB outperform NCBT and NCBB. Given the same number of measurements, the Riemannian method ROBB has the largest successful recovery probability among the four methods.



Figure 1: Empirical phase transition curves for 1000 random runs. The algorithms NCBT, NCBB, AMA, and ROBB are introduced in the first paragraph of Section 5.2.

5.4 Noisy Measurements

Figure 2 shows the relationships between RMSE and SNR for Algorithm 4 with L = 500 and 1000, and K = N = 100. The algorithm stops when the norm of the initial gradient over the norm of the last gradient is less than 10^{-12} . The noise measurements y are given by $\mathcal{A}(h_{\sharp}m_{\sharp}^{*}) + e$, where the noise e is $\frac{\tau w}{\|w\|_2 \|\mathcal{A}(h_{\sharp}m_{\sharp}^{*})\|_2}$ for some positive value τ . Clearly, increasing the number of measurements improves the accuracy, i.e., reduces the RMSE. In addition, the curves indicate that inreasing SNR in dB reduces the RMSE in dB linearly.



Figure 2: Performance of Algorithm 2 under different SNR.

5.5 Natural Image

A leaf image (Figure 3) with 1024×1024 pixels from FLAVIA dataset [WBX⁺07] is used to test the performance of Algorithm 3 on an image deblurring problem.



Figure 3: The tested image

A blurred image is a convolution of the original image with a blurring kernel. Let \mathbf{y} , $\hat{\mathbf{y}}$ and κ denote the vectors obtained by reshaping the blurred image, original image and the blurring kernel respectively. The measurement y is therefore the vector \mathbf{Fy}/\sqrt{L} . The matrix \mathbf{B} is formed by a subset of columns of a reshaped 2D frequency Fourier matrix, where the columns correspond to the nonzero entries in κ . Since most natural images are approximately sparse in Haar wavelet basis, the columns of $\mathbf{C} \in \mathbb{C}^{L \times N}$ are set as the N most-significant columns in Haar wavelet matrix \mathbf{W} , where the N most-significant columns denote the columns corresponding to the N largest coefficients of Haar wavelet transform of the original image (the N largest entries in $\mathbf{W}^T \hat{\mathbf{y}}$). However, the original image, or equivalently $\hat{\mathbf{y}}$, is unknown. Thus, we use $\mathbf{W}^T \mathbf{y}$ instead to form \mathbf{C} .

The number of measurements L is 1024 * 1024 = 1048576. The number of columns N in C is chosen to be 20000. The number of nonzero entries in a blurring kernel is given later.

Motion kernel with known support: The motion kernel⁷ and the corresponding blurred image are shown in (a) and (b) of Figure 4. We assume that the support of the blurring kernel is known.

⁷The blurring kernel is obtained by Matlab commands: "fspecial('motion', 50, 45)".



Figure 4: Left: the blurring kernel; middle: the blurred image; right: the reconstructed image

The number of nonzero entries in the blurring kernel is K = 109. The most-significant wavelet subspaces with N = 20000 are able to capture 97% of the energy in the original image. Algorithm 3 stops when the number of iterations is 20, 40, 60, 80, 100, 120, 140, 160, or 2000. The reconstructed image using 80 iterations is given in (c) of Figure 4. As shown in Table 2, a higher accuracy does not improve the recovery performance in the sense that the relative error *relerr*, defined in Table 2, does not necessarily decrease as the number of iterations increases. In the later experiments, the number of iterations is set to be 80.

Table 2: The computational costs for multiple values of N in Figure 4. nBh, nCm, and nFFT are defined in Table 1. t denotes the computational time in seconds. relres denotes $\|y - \operatorname{diag}(Bhm^*C^*)\|_2/\|y\|_2$. rellerr denotes $\|\hat{\mathbf{y}} - \frac{\|\mathbf{y}\|}{\|\mathbf{y}_f\|}\mathbf{y}_f\|/\|\hat{\mathbf{y}}\|$, where \mathbf{y}_f denotes the vector obtained by reshaping a reconstructed image.

num. of iter.	20	40	60	80	100	120	140	160	2000
relres	3.9_{-3}	2.6_{-3}	2.5_{-3}	2.4_{-3}	2.3_{-3}	2.3_{-3}	2.2_{-3}	2.2_{-3}	2.0_{-3}
nBh/nCm	43	85	125	168	210	253	294	335	4055
nFFT	107	211	311	417	521	627	729	831	10111
relerr	4.1_{-2}	3.8_{-2}	3.8_{-2}	3.8_{-2}	4.1_{-2}	4.2_{-2}	4.3_{-2}	4.4_{-2}	6.4_{-2}

Other kernels with known supports: Two blurring kernels, their corresponding blurred images, and the reconstructed images are shown in Figures 5 and 6. The blurring kernel in Figure 5 is from the function sin and has 153 nonzero entries. The covariance of the Gaussian kernel in Figure 6 is $V = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix}$ and the number of nonzero entries is 181.

The number of iterations is 80 and their computational times are approximately 48 seconds. The relative errors *relerr*, defined in Table 2, of the "sin" kernel and the Gaussian kernel are 0.0398 and 0.0890, respectively. We can see that the Riemannian method is able to recover a reasonable image for a more complex kernel.



Figure 5: Left: the blurring kernel; middle: the blurred image; right: the reconstructed image



Figure 6: Left: the blurring kernel; middle: the blurred image; right: the reconstructed image

Motion kernel with unknown support: The motion kernel in (a) of Figure 4 is used. In practice, the true support of the kernel is unknown and an estimation is usually not exact. Figure 7 shows the recovery performance of the Riemannian method when an inexact support is used. We use four inexact supports, which are obtained by enlarging the true support by 1, 2, 3, and 4 pixels. Unsurprisingly, the better the estimation of the support is, the better the reconstructed image is. The Riemannian method is able to recover the image reasonably in these tests.



Figure 7: Deconvolution results for unknown supports. Top: the estimated support of the blurring kernel. Bottom: the reconstructed images using the corresponding above supports. The relative error *relerr* from left to right are 0.0440, 0.0484, 0.0522, and 0.0673, respectively.

6 Conclusion

In this paper, we proposed a Riemannian steepest descent method for blind deconvolution. By inspiring the proofs in [LLSW16], it is proven that the Riemannian method with an appropriate initialization has high probability to successfully recover the desired signal. Since a quotient manifold is considered, the penalty term that is used to control the norm of h and m is not necessary. The Hessian at the desired solution is proven to be well-conditioned with high probability and therefore the optimization is not difficult locally. Numerical experiments show that the Riemannian steepest descent method is robust to noise, is the most efficient algorithm, and has the empirical largest successful recovery probability among the alternating minimization algorithm and the algorithm in [LLSW16].

References

- [AMS08] P.-A. Absil, R. Mahony, and R. Sepulchre. Optimization algorithms on matrix manifolds. Princeton University Press, Princeton, NJ, 2008.
- [ARR14] A. Ahmed, B. Recht, and J. Romberg. Blind deconvolution using convex programming. *IEEE Transactions on Information Theory*, 60(3):1711–1732, March 2014.

- [BA11] N. Boumal and P.-A. Absil. RTRMC: A Riemannian trust-region method for low-rank matrix completion. Advances in Neural Information Processing Systems 24 (NIPS), pages 406–414, 2011.
- [Bak08] C. G. Baker. Riemannian manifold trust-region methods with applications to eigenproblems. PhD thesis, Florida State University, Department of Computational Science, 2008.
- [BB88] J. Barzilai and J. M. Borwein. Two-Point Step Size Gradient Methods. IMA Journal of Numerical Analysis, 8:141–148, 1988.
- [BM03] S. Burer and R. D. C. Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, February 2003. doi:10.1007/s10107-002-0352-8.
- [BSA13] N. Boumal, A. Singer, and P.-A. Absil. Robust estimation of rotations from relative measurements by maximum likelihood. 52nd IEEE Conference on Decision and Control, (3):1156–1161, December 2013. doi:10.1109/CDC.2013.6760038.
- [CE07] P. Campisi and K. Egiazarian. Blind image deconvolution : theory and applications. CRC press, 2007.
- [CLS16] E. J. Candés, X. Li, and M. Soltanolkotabi. Phase retrieval via Wirtinger flow: theory and algorithms. *IEEE Transactions on Information Theory*, 64(4):1985–2007, 2016.
- [FJ05] Matteo Frigo and Steven G. Johnson. The design and implementation of FFTW3. Proceedings of the IEEE, 93(2):216–231, 2005. Special issue on "Program Generation, Optimization, and Platform Adaptation".
- [GV96] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, third edition, 1996.
- [HAG15] W. Huang, P.-A. Absil, and K. A. Gallivan. A Riemannian symmetric rank-one trustregion method. *Mathematical Programming*, 150(2):179–216, February 2015.
- [HAG16] W. Huang, P.-A. Absil, and K. A. Gallivan. Intrinsic representation of tangent vectors and vector transport on matrix manifolds. *Numerische Mathematik*, 136(2):523–543, 2016.
- [HAG18] Wen Huang, P.-A. Absil, and K. A. Gallivan. A Riemannian BFGS Method without Differentiated Retraction for Nonconvex Optimization Problems. SIAM Journal on Optimization, 28(1):470–495, 2018.
- [HAGH16] Wen Huang, P.-A. Absil, K. A. Gallivan, and Paul Hand. ROPTLIB: an object-oriented C++ library for optimization on Riemannian manifolds. Technical Report FSU16-14, Florida State University, 2016.
- [HGA15] W. Huang, K. A. Gallivan, and P.-A. Absil. A Broyden Class of Quasi-Newton Methods for Riemannian Optimization. SIAM Journal on Optimization, 25(3):1660–1685, 2015.

- [HGZ17] Wen Huang, K. A. Gallivan, and Xiangxiong Zhang. Solving PhaseLift by low rank Riemannian optimization methods for complex semidefinite constraints. SIAM Journal on Scientific Computing, 39(5):B840–B859, 2017.
- [Hua13] W. Huang. Optimization algorithms on Riemannian manifolds with applications. PhD thesis, Florida State University, Department of Mathematics, 2013.
- [JC93] S. M. Jefferies and J. C. Christou. Restoration of astronomical images by iterative blind deconvolution. *The Astrophysical Journal*, page 415:862, 1993.
- [LLJB15] K. Lee, Y. Li, M. Junge, and Y. Bresler. Blind Recovery of Sparse Signals from Subsampled Convolution. pages 1–43, 2015.
- [LLSW16] Xiaodong Li, Shuyang Ling, Thomas Strohmer, and Ke Wei. Rapid, robust, and reliable blind deconvolution via nonconvex optimization. *CoRR*, abs/1606.04933, 2016.
- [LS15] S. Ling and T. Strohmer. Self-Calibration and Biconvex Compressive Sensing. Inverse Problems, 31(11):115002, 2015.
- [LWDF11] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2354–2367, 2011.
- [Mis14] B. Mishra. A Riemannian approach to large-scale constrained least-squares with symmetries. PhD thesis, University of Liege, 2014.
- [NW06] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, second edition, 2006.
- [RW12] W. Ring and B. Wirth. Optimization methods on Riemannian manifolds and their application to shape space. SIAM Journal on Optimization, 22(2):596–627, January 2012. doi:10.1137/11082885X.
- [Sat16] H. Sato. A Dai–Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions. *Computational Optimization and Applications*, 64(1):101–118, May 2016.
- [SI15] H. Sato and T. Iwai. A new, globally convergent Riemannian conjugate gradient method. *Optimization*, 64(4):1011–1031, February 2015.
- [SL16] Ruoyu Sun and Zhi Quan Luo. Guaranteed Matrix Completion via Non-Convex Factorization. *IEEE Transactions on Information Theory*, 62(11):6535–6579, 2016.
- [Van13] B. Vandereycken. Low-rank matrix completion by Riemannian optimization—extended version. *SIAM Journal on Optimization*, 23(2):1214–1236, 2013.
- [WBSJ15] G. Wunder, H. Boche, T. Strohmer, and P. Jung. Sparse Signal Processing Concepts for Efficient 5G System Design. *IEEE Access*, 3:195–208, 2015.
- [WBX⁺07] S. G. Wu, F. S. Bao, E. Y. Xu, Y.-X. Wang, Y.-F. Chang, and Q.-L. Xiang. A leaf recognition algorithm for plant classification using probabilistic neural network. 2007 IEEE International Symposium on Signal Processing and Information Technology, pages 11–16, 2007. arXiv:0707.4289v1.

- [WCCL16] K. Wei, J.-F. Cai, T. F. Chan, and S. Leung. Guarantees of Riemannian Optimization for Low Rank Matrix Completion. (1), 2016.
- [WP98] X. Wang and H. V. Poor. Blind Equalization and Multiuser Detection in Dispersive CDMA Channels. *IEEE Transactions on Communications*, 46(1):91–103, 1998.
- [YK94] T.-L. You and M. Kaveh. A simple algorithm for joint blur identification and image restoration. Proceedings - International Conference on Image Processing, ICIP, 3(3):167–171, 1994.
- [Zhu17] X. Zhu. A Riemannian conjugate gradient method for optimization on the Stiefel manifold. *Computational Optimization and Applications*, 67(1):73–110, 2017.

A Four Conditions and the Proof of Theorem 3.2

The convergence analysis of the Riemannian steepest descent method follows the spirit of the analysis in [LLSW16] and the analyses both rely on the four conditions: local RIP condition, robustness condition, local regularity condition and local smoothness condition. There exist differences and the main ones are highlighted as follows. The differences ease the proofs of the Riemannian method in general.

- The cost function \tilde{f} does not include the penalty terms for the norm of h and m. Therefore, the penalty terms are not considered in the convergence analysis.
- Since any representation in $\pi^{-1}(\pi(h, m))$ can be used and this does not influence the sequence $\{\pi(h_k, m_k)\}$ of \mathcal{Q}_1 generated by Algorithm 4, we can always assume $\|h_k\|_2 = \|m_k\|_2$ without loss of generality.
- The Riemannian gradient is different from the Wirtinger derivative. For the cost function F in (1.4), if $||h||_2 = ||m||_2$, then by (4.9) and (3.4), we have

$$\operatorname{grad} F(h,m) = \frac{2}{\|hm^*\|_2} \nabla^w F(h,m).$$

Let $\pi(h_{\sharp}, m_{\sharp})$ denote the ground truth and $\Upsilon_{\tilde{f}}$ denote $\{\pi(h, m) \mid \tilde{f}(\pi(h, m)) \leq \frac{1}{3}\varepsilon^2 d_*^2 + \|e\|_2^2\}$.

Condition A.1 (Local RIP condition). The following local Restricted Isometry Property for \mathcal{A} holds uniformly for all $(h, m) \in \Pi_{\varepsilon}$:

$$\frac{3}{4} \|hm^* - h_{\sharp}m_{\sharp}^*\|_F^2 \le \|\mathcal{A}(hm^* - h_{\sharp}m_{\sharp}^*)\|_2^2 \le \frac{5}{4} \|hm^* - h_{\sharp}m_{\sharp}^*\|_F^2.$$

Condition A.2 (Robustness condition). If $L \ge C_{\gamma}(\sigma^2/\varepsilon^2 + \sigma/\varepsilon) \max(K, N) \log(L)$, then with high probability, it holds that

$$\|\mathcal{A}^*(e)\|_2 \le \frac{\varepsilon d_*}{10\sqrt{2}}$$

Condition A.3 (Local regularity condition). There exists a regularity constant $a_0 > 0$ such that

$$\| \operatorname{grad} \tilde{f}(\pi(h,m)) \|_{g_{\pi(h,m)}}^2 \ge a_0[\tilde{f}(\pi(h,m)) - c]_+$$

for all $\pi(h,m) \in \Omega_{\mu} \cap \Pi_{\varepsilon}$, where $c = \|e\|_2^2 + 1700 \|\mathcal{A}^*(e)\|_2^2$ and $a_0 = 1/1500$.

Condition A.4 (Local smoothness condition). Define the lifting function $\hat{f}_{\pi(h,m)} : T_{\pi(h,m)} \mathcal{Q}_1 \to \mathbb{R} : \eta_{\pi(h,m)} = \tilde{f} \circ \tilde{R}_{\pi(h,m)}(\eta_{\pi(h,m)})$. There exists a constant a_L such that

 $\|\operatorname{grad} \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)}) - \operatorname{grad} \hat{f}_{\pi(h,m)}(0)\|_{g_{(h,m)}} \le a_L t \|\eta_{\pi(h,m)}\|_{g_{(h,m)}}, \quad \forall 0 \le t \le 1,$

for all $\pi(h,m)$ and $\eta_{\pi(h,m)} \in T_{\pi(h,m)} \mathcal{Q}_1$ such that $\tilde{R}_{\pi(h,m)}(t\eta_{\pi(h,m)}) \in \Upsilon_{\tilde{f}} \cap \Pi_{\varepsilon}, \forall 0 \leq t \leq 1$, where \tilde{R} is defined in (4.5).

Conditions A.1 and A.2 are the same as [LLSW16, Conditions 5.1 and 5.2] and have been proven therein. Conditions A.3 and A.4 are different from[LLSW16, Conditions 5.3 and 5.4] since we use different penalty term and different gradient and norm. The proofs of Conditions A.3 and A.4 are given in Section B.

Lemma A.1 generalizes [LLSW16, Lemma 6.1] and is used in Lemma A.2.

Lemma A.1 (Riemannian descent lemma). Suppose Condition A.4 holds. Then

$$\hat{f}_{\pi(h,m)}(\eta_{\pi(h,m)}) \le \hat{f}(0_{\pi(h,m)}) + g_{\pi(h,m)}(\eta_{\pi(h,m)}, \operatorname{grad} \tilde{f}(\pi(h,m))) + a_L \|\eta_{\pi(h,m)}\|_g^2,$$

where $0_{\pi(h,m)}$ denotes the origin of $T_{\pi(h,m)} \mathcal{Q}_1$.

Proof. Define $\beta(t) = \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)})$. We have $\frac{d}{dt}\beta(t) = g\left(\operatorname{grad} \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)}), \eta_{\pi(h,m)}\right)$. It follows from the Fundamental Theorem of Calculus that

$$\begin{split} \hat{f}_{\pi(h,m)}(\eta_{\pi(h,m)}) &- \hat{f}(0) = \int_{0}^{1} \frac{d}{dt} \beta(t) dt = \int_{0}^{1} g\left(\operatorname{grad} \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)}), \eta_{\pi(h,m)} \right) dt \\ &= g\left(\operatorname{grad} \hat{f}_{\pi(h,m)}(0_{\pi(h,m)}), \eta_{\pi(h,m)} \right) + \int_{0}^{1} g\left(\operatorname{grad} \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)}) - \operatorname{grad} \hat{f}_{\pi(h,m)}(0_{\pi(h,m)}), \eta_{\pi(h,m)} \right) dt \\ &\leq g\left(\operatorname{grad} \hat{f}_{\pi(h,m)}(0_{\pi(h,m)}), \eta_{\pi(h,m)} \right) + a_L \|\eta_{\pi(h,m)}\|_g^2. \end{split}$$

Therefore, the result holds since grad $\hat{f}_{\pi(h,m)}(0_{\pi(h,m)}) = \operatorname{grad} \tilde{f}(\pi(h,m))$, see [AMS08, (4.4)].

Lemma A.2 is used in the proof of Theorem 3.2. Note that this lemma follows from [LLSW16, Section 5.1].

Lemma A.2. The following properties hold under some of the four conditions.

1. Under conditions A.1 and A.2, function f in (1.5) satisfies

$$\|e\|_{2}^{2} + \frac{3}{4}\Delta^{2} - \frac{\varepsilon\Delta d_{*}}{5} \le f(\pi(h,m)) \le \|e\|_{2}^{2} + \frac{5}{4}\Delta^{2} + \frac{\varepsilon\Delta d_{*}}{5}$$
(A.1)

for all $\pi(h,m) \in \Omega_{\mu} \cap \Pi_{\varepsilon}$, where $\Delta = \|hm^* - h_{\sharp}m_{\sharp}^*\|_F$.

- 2. It holds that $\Upsilon_{\tilde{f}} \subset \Omega_{\mu}$; under conditions A.1 and A.2, we have $\Upsilon_{\tilde{f}} \cap \Pi_{\varepsilon} \subset \Pi_{\frac{9}{20}\varepsilon}$.
- 3. Under conditions A.1 and A.2, if $\pi(h_1, m_1) \in \Pi_{\varepsilon}$ and $\pi((1 \lambda)h_1 + \lambda h_2, (1 \lambda)m_1 + \lambda m_2) \in \Upsilon_{\tilde{f}}$ for all $\lambda \in [0, 1]$, then $\pi(h_2, m_2) \in \Pi_{\varepsilon}$.

4. Under conditions A.1, A.2 and A.4, suppose the step size $\alpha \leq \frac{1}{2a_L}$, where a_L is defined in Condition A.4 and $\pi(h_k, m_k) \in \Pi_{\varepsilon} \cap \Upsilon_{\tilde{f}}$, then it holds that $\pi(h_{k+1}, m_{k+1}) \in \Pi_{\varepsilon} \cap \Upsilon_{\tilde{f}}$ and

$$\tilde{f}(\pi(h_{k+1}, m_{k+1})) \le \tilde{f}(\pi(h_k, m_k)) - \frac{\alpha}{2} \| \operatorname{grad} \tilde{f}(\pi(h_k, m_k)) \|_g^2$$

Proof. (1): This has been proven in [LLSW16, (5.5), (5.6)].

(2): If $\pi(h,m) \notin \Omega_{\mu}$, then $G(\pi(h,m)) \ge \rho G_0(\frac{2d_*^2}{d^2})$. It follows that

$$\begin{split} \tilde{f}(\pi(h,m)) \geq &\rho G_0(\frac{2d_*^2}{d^2}) \geq (d^2 + 2.5 \|e\|_2^2) \left(\frac{2d_*^2}{d^2} - 1\right)^2 \\ \geq & d^2 \left(\frac{2d_*^2}{d^2} - 1\right)^2 + 2.5 \|e\|_2^2 \left(\frac{2d_*^2}{d^2} - 1\right)^2 \geq \frac{1}{3}\varepsilon^2 d_*^2 + \|e\|_2^2 \end{split}$$

,

where $\rho \geq d^2 + 2.5 \|e\|_2^2$, $\varepsilon < 1/15$, and $0.9d_* \leq d \leq 1.1d_*$. Therefore, $\pi(h,m) \notin \Upsilon_{\tilde{f}}$ and $\Upsilon_{\tilde{f}} \subset \Omega_{\mu}$.

For any $\pi(h,m) \in \Upsilon_{\tilde{f}} \cap \Pi_{\varepsilon}$, it holds that $\pi(h,m) \in \Omega_{\mu} \cap \Pi_{\varepsilon}$. It follows from (A.1) that $\|e\|_{2}^{2} + \frac{3}{4}\Delta^{2} - \frac{\varepsilon\Delta d_{*}}{5} \leq f(\pi(h,m)) \leq \tilde{f}(\pi(h,m)) \leq \frac{1}{3}\varepsilon^{2}d_{*}^{2} + \|e\|_{2}^{2}$, which can be simplified into $45\Delta^{2} - 12\varepsilon d_{*}\Delta - 20\varepsilon^{2}d_{*}^{2} \leq 0$. This implies $\Delta \leq 0.9\varepsilon d_{*}$ and hence $\Upsilon_{\tilde{f}} \cap \Pi_{\varepsilon} \subset \Pi_{\frac{9}{10}\varepsilon}$.

(3): Prove by contradiction. If $\pi(h_2, m_2) \notin \Pi_{\varepsilon}$, then there exists λ_0 such that

$$\| ((1-\lambda)h_1 + \lambda h_2) ((1-\lambda)m_1 + \lambda m_2)^* - h_1 m_1^* \| = \varepsilon d_*,$$

i.e., $\pi ((1 - \lambda)h_1 + \lambda h_2, (1 - \lambda)m_1 + \lambda m_2) \in \partial \Pi_{\varepsilon}$. Therefore, it follows from

$$\pi\left((1-\lambda)h_1+\lambda h_2,(1-\lambda)m_1+\lambda m_2\right)\in\Upsilon_{\tilde{f}}$$

that $\pi((1-\lambda)h_1 + \lambda h_2, (1-\lambda)m_1 + \lambda m_2) \in \Pi_{0.9\varepsilon}$. This is a contradiction.

(4): If grad $\tilde{f}(\pi(h_k, m_k)) = 0$, then $\pi(h_{k+1}, m_{k+1}) = \pi(h_k, m_k) \in \Pi_{\varepsilon} \cap \Upsilon_{\tilde{f}}$. Suppose grad $\tilde{f}(\pi(h_k, m_k)) \neq 0$. Define the function

$$\phi(\lambda) = \hat{f}_{\pi(h_k, m_k)}(-\lambda \operatorname{grad} \tilde{f}(\pi(h_k, m_k))).$$

It follows that $\phi'(0) = -\| \operatorname{grad} \tilde{f}(\pi(h_k, m_k)) \|_q^2 < 0$. By Lemma A.1, we have

$$\phi(\lambda) \le \phi(0) - \lambda \|\operatorname{grad} \tilde{f}(\pi(h_k, m_k))\|_g^2 + a_L \lambda^2 \|\operatorname{grad} \tilde{f}(\pi(h_k, m_k))\|_g^2 \tag{A.2}$$

$$=\phi(0) + (a_L \lambda^2 - \lambda) \| \operatorname{grad} \tilde{f}(\pi(h_k, m_k)) \|_g^2 \le \phi(0)$$
(A.3)

for all $\lambda \in [0, \alpha]$. Therefore, by (3) in Lemma A.2, we have $\pi(h_{k+1}, m_{k+1}) \in \Pi_{\varepsilon} \cap \Upsilon_{\tilde{f}}$. Using (A.3) yields

$$\tilde{f}(\pi(h_{k+1}, m_{k+1})) \le \tilde{f}(\pi(h_k, m_k)) - \frac{\alpha}{2} \| \operatorname{grad} \tilde{f}(\pi(h_k, m_k)) \|_g^2.$$

Now, we are ready to prove Theorem 3.2. With the four conditions, the proof follows from the proof in [LLSW16] and we give here for completeness.

Proof. Since $\pi(h_0, m_0) \in \Omega_{\frac{1}{2}\mu} \cap \prod_{\frac{2}{5}\varepsilon}$, we have

$$\frac{L|b_i^*h_0|^2||m_0||_2^2}{8d^2\mu^2} \le \frac{L}{8d^2\mu^2} \frac{4d_*^2\mu^2}{L} \le \frac{d_*^2}{2d^2} < 1,$$

where $\sqrt{L} \|Bh_0\|_{\infty} \|m\|_2 \leq 2d_*\mu$. Therefore, the penalty term $G(\pi(h_0, m_0)) = 0$. Combining (A.1) and $\Delta \leq \frac{2}{5}\varepsilon d_*$ yields

$$\tilde{f}(\pi(h,m)) = f(\pi(h,m)) \le \|e\|_2^2 + \frac{5}{4}\Delta^2 + \frac{\varepsilon\Delta d_*}{5} \le \frac{1}{3}\varepsilon^2 d_*^2 + \|e\|_2^2,$$

which implies $\pi(h_0, m_0) \in \Upsilon_{\tilde{f}}$. It follows from (4) in Lemma A.2 and Condition A.3 that

$$\tilde{f}(\pi(h_{k+1}, m_{k+1})) \le \tilde{f}(\pi(h_k, m_k)) - \frac{\alpha a_0}{2} \left[\tilde{f}(\pi(h_k, m_k)) - c \right]_+$$

and $\pi(h_k, m_k) \in \Omega_{\mu} \cap \Pi_{\varepsilon}$ for all k. Therefore, we have

$$\tilde{f}(\pi(h_{k+1}, m_{k+1})) - c \le \left(1 - \frac{\alpha a_0}{2}\right) \left[\tilde{f}(\pi(h_k, m_k)) - c\right]_+$$

which implies

$$\left[\tilde{f}(\pi(h_{k+1}, m_{k+1})) - c\right]_{+} \le \left(1 - \frac{\alpha a_0}{2}\right) \left[\tilde{f}(\pi(h_k, m_k)) - c\right]_{+}.$$

Thus, we obtain

$$\left[\tilde{f}(\pi(h_k, m_k)) - c\right]_+ \le \left(1 - \frac{\alpha a_0}{2}\right)^k \left[\tilde{f}(\pi(h_0, m_0)) - c\right]_+ \le \frac{1}{3} \left(1 - \frac{\alpha a_0}{2}\right)^k \varepsilon^2 d_*^2,$$

where we use $\tilde{f}(\pi(h_0, m_0)) \leq \frac{1}{3}\varepsilon^2 d_*^2 + \|e\|_2^2$ and $c = \|e\|_2^2 + 1700\|\mathcal{A}^*(e)\|_2^2 \geq \|e\|_2^2$. We also have

$$\begin{split} \tilde{f}(\pi(h_k, m_k)) - \|e\|_2^2 \geq & \|\mathcal{A}(h_k m_k^* - h_\sharp m_\sharp^*)\|_2^2 - 2\operatorname{Re}(\left\langle \mathcal{A}^*(e), h_k m_k^* - h_\sharp m_\sharp^*\right\rangle_2) \\ \geq & \frac{3}{4} \|h_k m_k^* - h_\sharp m_\sharp^*\|_F^2 - 2\sqrt{2} \|\mathcal{A}^*(e)\|_2 \|h_k m_k^* - h_\sharp m_\sharp^*\|_F. \end{split}$$

It follows that

which is equivalent to

$$\left| \|h_k m_k^* - h_{\sharp} m_{\sharp}^* \|_F - \frac{4\sqrt{2}}{3} \|\mathcal{A}^*(e)\|_2 \right|^2 \le \frac{4}{9} \left(1 - \frac{\alpha a_0}{2} \right)^t \varepsilon^2 d_*^2 + \left(\frac{6800}{3} + \frac{32}{9} \right) \|\mathcal{A}^*(e)\|_2^2.$$

Solving for $||h_k m_k^T - h_{\sharp} m_{\sharp}^T||_F$ yields

$$\|h_k m_k^T - h_{\sharp} m_{\sharp}^T\|_F \le \frac{2}{3} \left(1 - \frac{\alpha a_0}{2}\right)^{k/2} \varepsilon d_* + 50 \|\mathcal{A}^*(e)\|_2$$

The upper bound for $\|\mathcal{A}^*(e)\|_2$ has been proven in [LLSW16].

B Proofs of Conditions A.3 and A.4 and Theorem 3.1

Define function $\tilde{f}_{\mathrm{T}} = \tilde{f} \circ \pi$, $f_{\mathrm{T}} = f \circ \pi$, and $G_{\mathrm{T}} = G \circ \pi$, which is defined in the total space $\mathbb{R}^{K}_{*} \times \mathbb{R}^{N}_{*}$. Since the function value $f_{\mathrm{T}}(h,m)$ and $\| \operatorname{grad} f_{\mathrm{T}}(h,m) \|_{g}$ are independent of representation in $\pi^{-1}(\pi(h,m))$, we can always choose h and m such that $\|h\|_{2} = \|m\|_{2}$. For all $\pi(h,m) \in \Pi_{\varepsilon}$, we have $\|h\|_{2}\|m\|_{2} = \|hm^{*}\|_{F} \leq \|hm^{*} - h_{\sharp}m^{*}_{\sharp}\|_{F} + \|h_{\sharp}m^{*}_{\sharp}\|_{F} \leq (1+\varepsilon)d_{*}$ and $\|h\|_{2}\|m\|_{2} = \|hm^{*}\|_{F} \geq \|hm^{*} - h_{\sharp}m^{*}_{\sharp}\|_{F} \geq (1-\varepsilon)d_{*}$. It follows that

$$(h,m) \in \Omega_d := \left\{ (h,m) \mid \sqrt{\frac{14}{15}} d_* \le \|h\|_2 = \|m\|_2 \le \sqrt{\frac{16}{15}} d_* \right\},\tag{B.1}$$

for all $\pi(h,m) \in \Pi_{\varepsilon}$ and $\varepsilon \leq 1/15$. There are unique orthogonal decompositions

$$h = \tau_1 h_{\sharp} + \tilde{h}$$
 and $m = \tau_2 m_{\sharp} + \tilde{m}$,

where $\tilde{h}^* h_{\sharp} = 0$ and $\tilde{m}^* m_{\sharp} = 0$. Let

$$\hat{h} = h - \tau h_{\sharp}$$
 and $\hat{m} = m - \bar{\tau}^{-1} m_{\sharp}$,

where $\tau = \frac{1}{(1 - \frac{\Delta}{10d_*})\bar{\tau}_2}$.

Lemma B.1. For all (h,m) such that $(h,m) \in \Omega_d$ and $\pi(h,m) \in \Pi_{\varepsilon}$ with $\varepsilon \leq 1/15$, it holds that $\|\hat{h}\|_2^2 \leq 6.1\Delta^2/d_*$, $\|\hat{m}\|_2^2 \leq 6.1\Delta^2/d_*$, and $\|\hat{h}\|_2^2 \|\hat{m}\|_2^2 \leq 8.4\Delta^4/d_*^2$. Moreover, if we assume $\pi(h,m) \in \Omega_{\mu}$ additionally, we have $\sqrt{L} \|B(\hat{h})\|_{\infty} \leq 6\mu\sqrt{d_*}$.

Proof. Inequalities $\|\hat{h}\|_{2}^{2} \leq 6.1\Delta^{2}/d_{*}$, $\|\hat{m}\|_{2}^{2} \leq 6.1\Delta^{2}/d_{*}$, and $\|\hat{h}\|_{2}^{2}\|\hat{m}\|_{2}^{2} \leq 8.4\Delta^{4}/d_{*}^{2}$ have been proven in [LLSW16, Lemma 5.15]. Note the definition of Ω_{μ} and (B.1), we have $\{(h,m) \mid \sqrt{L}\|Bh\|_{\infty}\|m\|_{2} \leq 4d_{*}\mu, \|h\|_{2} = \|m\|_{2}\} \subset \{(h,m) \mid \sqrt{L}\|Bh\|_{\infty} \leq 4.5\sqrt{d_{*}}\mu\}$. We also have $|\tau_{1}\bar{\tau_{2}}| \geq 1-\varepsilon$ by [LLSW16, Lemma 5.9] and $|\tau_{1}| \leq \frac{\|h\|_{2}}{\|h_{\sharp}\|_{2}} = \sqrt{\frac{\|h\|_{2}\|m\|_{2}}{\|h_{\sharp}\|_{2}}} \leq \sqrt{\frac{\|hm^{*}-h_{\sharp}m_{\sharp}^{*}\|_{F}+\|h_{\sharp}m_{\sharp}^{*}\|_{F}}{d_{*}}} \leq \sqrt{1+\varepsilon}$. Therefore, it holds that

$$\begin{split} \sqrt{L} \|B(\hat{h})\|_{\infty} &\leq \sqrt{L} \|B(h)\|_{\infty} + \frac{1}{(1 - \frac{\Delta}{10d_*})\bar{\tau}_2} \sqrt{L} \|B(h_{\sharp})\|_{\infty} \\ &= 4.5\mu\sqrt{d_*} + \frac{\sqrt{1 + \varepsilon}}{(1 - \frac{\Delta}{10d_*})(1 - \varepsilon)} \leq 5.7\mu\sqrt{d_*}, \end{split}$$

where $\Delta/d_* \leq \varepsilon \leq 1/15$.

Lemma 2.3 is the [LLSW16, Lemm 5.16] and is used in the proof of Condition A.3.

Lemma B.2. For all (h,m) such that $(h,m) \in \Omega_d$ and $\pi(h,m) \in \Pi_{\varepsilon} \cap \Omega_{\mu}$ with $\varepsilon \leq 1/15$, it uniformly holds that

$$\operatorname{Re}\left(\left\langle \nabla_{h}^{w} f_{\mathrm{T}}, \hat{h} \right\rangle_{2} + \left\langle \nabla_{m}^{w} f_{\mathrm{T}}, \hat{m} \right\rangle_{2}\right) \geq \frac{\Delta^{2}}{8} - 2\Delta \|\mathcal{A}^{*}(e)\|_{2},$$

provided $L \ge a_3 \mu^2 (K+N) \log^2(L)$ for some constant a_3 , where ∇^w denotes the Wirtinger derivative (3.4).

1		
1		
1		

Lemma B.3. For all (h,m) such that $(h,m) \in \Omega_d$ and $\pi(h,m) \in \Pi_{\varepsilon}$ with $\varepsilon \leq 1/15$ and $\frac{9}{10}d_* \leq d \leq \frac{11}{10}d_*$, it uniformly holds that

$$\operatorname{Re}\left(\left\langle \nabla_{h}^{w}G_{\mathrm{T}},\hat{h}\right\rangle_{2}+\left\langle \nabla_{m}^{w}G_{\mathrm{T}},\hat{m}\right\rangle_{2}\right)\geq\frac{\Delta}{5d_{*}}\sqrt{\rho G(\pi(h,m))},$$

where $\rho \ge d^2 + 2.5 \|e\|_2^2$.

Proof. We have

$$\left\langle \nabla_h G_{\mathrm{T}}, \hat{h} \right\rangle_2 + \left\langle \nabla_m G_{\mathrm{T}}, \hat{m} \right\rangle_2 = \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^L G_0' \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) \left\langle b_i b_i^*h, \hat{h} \right\rangle_2 ||m||_2^2 + \frac{L\rho}{4d^2\mu^2} \sum_{i=1}^L G_0' \left(\frac{L|b_i^*h|^2 ||m||_2^2}{8d^2\mu^2} \right) \left\langle m, \hat{m} \right\rangle_2 |b_i^*h|^2$$

When $L|b_{i}^{*}h|^{2}||m||_{2}^{2} \leq 8d^{2}\mu^{2}$, we have

$$\frac{L\rho}{4d^{2}\mu^{2}}\sum_{i=1}^{L}G_{0}^{\prime}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)\left\langle b_{i}b_{i}^{*}h,\hat{h}\right\rangle_{2}\|m\|_{2}^{2}=0=\frac{\rho}{5}G_{0}^{\prime}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right) \text{ and }$$
$$\frac{L\rho}{4d^{2}\mu^{2}}\sum_{i=1}^{L}G_{0}^{\prime}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)\left\langle m,\hat{m}\right\rangle_{2}|b_{i}^{*}h|^{2}=0=\frac{\rho}{5}G_{0}^{\prime}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)$$

When $L|b_i^*h|^2 ||m||_2^2 > 8d^2\mu^2$, by [LLSW16, Lemma 5.9] and (B.1), it holds that $|\tau_1 \bar{\tau}_2 - 1| \leq \frac{\Delta}{d_*}$, $|\tau_1| \leq 2$, which yields

$$\tau = \frac{1}{(1 - \frac{\Delta}{10d_*})|\bar{\tau}_2|} = \frac{|\tau_1|}{(1 - \frac{\Delta}{10d_*})|\tau_1\bar{\tau}_2|} \le \frac{2}{(1 - \frac{\Delta}{10d_*})(1 - \frac{\Delta}{d_*})}.$$

By $\mu_h \leq \mu$ and $\Delta/d_* \leq \varepsilon \leq 1/15$, we obtain

$$\begin{aligned} \operatorname{Re}(\left\langle b_{i}b_{i}^{*}h,\hat{h}\right\rangle_{2}) &= \operatorname{Re}(|b_{i}^{*}h|^{2} - \tau \langle b_{i}^{*}h,b_{i}^{*}h_{\sharp}\rangle_{2}) \\ &\geq |b_{i}^{*}h|(|b_{i}^{*}h| - \frac{2}{(1 - \frac{\Delta}{10d_{*}})(1 - \frac{\Delta}{d_{*}})}|b_{i}^{*}h_{\sharp}|) \\ &\geq |b_{i}^{*}h|(|b_{i}^{*}h| - \frac{2\mu}{(1 - \frac{\Delta}{10d_{*}})(1 - \frac{\Delta}{d_{*}})}\sqrt{\frac{d_{*}}{L}}) \text{ (by } |b_{i}^{*}h_{\sharp}| \leq \mu_{h}\sqrt{d_{*}/L}) \\ &\geq \sqrt{\frac{8d^{2}\mu^{2}}{L||m||_{2}^{2}}} \left(\sqrt{\frac{8d^{2}\mu^{2}}{L||m||_{2}^{2}}} - \frac{2\mu}{(1 - \frac{\Delta}{10d_{*}})(1 - \frac{\Delta}{d_{*}})}\sqrt{\frac{d_{*}}{L}}\right) \\ &\geq \sqrt{\frac{8d^{2}\mu^{2}}{L||m||_{2}^{2}}} \left(\sqrt{\frac{8d^{2}\mu^{2}}{L||m||_{2}^{2}}} - \frac{2\mu}{(1 - \frac{\Delta}{10d_{*}})(1 - \frac{\Delta}{d_{*}})}\sqrt{\frac{10d}{9L}}\frac{16}{15}\frac{10}{9}\frac{d}{||m||_{2}^{2}}}\right) \text{ (by (B.1) and } d_{*} \leq \frac{10}{9}d) \\ &\geq \frac{4d^{2}\mu^{2}}{5L||m||_{2}^{2}}. \end{aligned}$$

Therefore,

$$\frac{L\rho\|m\|_2^2}{4d^2\mu^2}G_0'\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right)\operatorname{Re}\left(\left\langle b_ib_i^*h,\hat{h}\right\rangle_2\right) \ge \frac{\rho}{5}G_0'\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right).$$

We also have

$$\operatorname{Re}(\langle m, \hat{m} \rangle_2) = \operatorname{Re}(\langle m, m - \bar{\tau}^{-1} m_{\sharp} \rangle_2) = \frac{\Delta}{10d_*} \|m\|_2^2 + \left(1 - \frac{\Delta}{10d_*}\right) \|\tilde{m}\|_2^2 \ge \frac{\Delta}{10d_*} \|m\|_2^2.$$

It follows that

$$\begin{aligned} \frac{L\rho}{4d^2\mu^2}G_0'\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right)\operatorname{Re}(\langle m,\hat{m}\rangle_2)|b_i^*h|^2 &\geq \frac{L\rho}{4d^2\mu^2}G_0'\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right)\frac{\Delta}{10d_*}\|m\|_2^2\frac{8d^2\mu^2}{L\|m\|_2^2}\\ &\geq \frac{\rho}{75}G_0'\left(\frac{L|b_i^*h|^2\|m\|_2^2}{8d^2\mu^2}\right).\end{aligned}$$

Therefore, we have

$$\operatorname{Re}\left(\left\langle \nabla_{h}G_{\mathrm{T}},\hat{h}\right\rangle_{2}+\left\langle \nabla_{m}G_{\mathrm{T}},\hat{m}\right\rangle_{2}\right)\geq\sum_{i=1}^{L}\frac{16\rho}{75}G_{0}'\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)$$
$$=\frac{32\rho}{75}\sum_{i=1}^{L}\sqrt{G_{0}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)}\geq\frac{32\rho}{75}\sqrt{\sum_{i=1}^{L}G_{0}\left(\frac{L|b_{i}^{*}h|^{2}\|m\|_{2}^{2}}{8d^{2}\mu^{2}}\right)}=\frac{32}{75}\sqrt{\rho G(\pi(h,m))}$$

The final result follows from the above inequality, (3.4) and $\Delta/d_* \leq 1/15$.

Proof of Condition A.3. Suppose (h,m) satisfies $(h,m) \in \Omega_d$ and $\pi(h,m) \in \Pi_{\varepsilon} \cap \Omega_{\mu}$ with $\varepsilon \leq 1/15$. Since Lemmas B.1, B.2 and B.3 are exactly the same as [LLSW16, Lemma 5.15, Lemma 5.16, and Lemma 5.17], we have the same result as [LLSW16, Lemma 5.18], i.e.,

$$\|\nabla^w \tilde{f}_{\mathrm{T}}(h,m)\|_2^2 \ge \frac{d_*}{5000} [\tilde{f}(\pi(h,m)) - c]_+$$

By Lemma 4.4 and (3.4), $||h||_2 = ||m||_2$ implies that

$$\nabla^{w}\tilde{f}_{\mathrm{T}}(h,m) = \frac{1}{2}P^{h}_{(h,m)}\nabla\tilde{f}_{\mathrm{T}}(h,m) = \frac{\|h\|_{2}\|m\|_{2}}{2}\left(\operatorname{grad}\tilde{f}(\pi(h,m))\right)_{\uparrow_{(h,m)}}$$

By definition of the induced-norm, we have

$$\|\operatorname{grad} \tilde{f}(\pi(h,m))\|_{g_{\pi(h,m)}}^{2} = \frac{4}{\|h\|_{2}\|m\|_{2}} \|\nabla^{w} \tilde{f}_{\mathrm{T}}(h,m)\|_{2}^{2}$$
$$\geq \frac{1}{1500} [\tilde{f}(\pi(h,m)) - c]_{+}. \tag{B.2}$$

Since $\| \operatorname{grad} \tilde{f}(\pi(h,m)) \|_{g_{\pi(h,m)}}^2$ and $\tilde{f}(\pi(h,m))$ are independent of representation in $\pi^{-1}(\pi(h,m))$, we have that (B.2) holds for all $\pi(h,m) \in \Pi_{\varepsilon} \cap \Omega_{\mu}$ with $\varepsilon \leq 1/15$. \Box Proof of Condition A.4. By [LLSW16, Condition 5.4], we have

$$\left\| \left(\nabla \tilde{f}_{\mathrm{T}}(t\eta_{\uparrow(h,m)}) \right) - \left(\nabla \tilde{f}_{\mathrm{T}}(0) \right) \right\|_{2} \leq \tilde{a}_{L} t \|\eta_{\uparrow(h,m)}\|_{2}, \quad \forall 0 \leq t \leq 1,$$

for some positive \tilde{a}_L . Choosing (h, m) such that $||h||_2 = ||m||_2$ yields that

$$\frac{14}{15}d_* \|\xi_{\uparrow(h,m)}\|_2^2 \le \|\xi_{\uparrow(h,m)}\|_g^2 \le \frac{16}{15}d_* \|\xi_{\uparrow(h,m)}\|_2^2,$$

and $\nabla \tilde{f}_{\mathrm{T}}(\xi_{\uparrow(h,m)}) = \|h\|_2 \|m\|_2 \left(\operatorname{grad} \hat{f}_{\pi(h,m)}(\xi_{\pi(h,m)}) \right)_{\uparrow_{(h,m)}}$ for all $\xi_{\pi(h,m)} \in \mathrm{T}_{\pi(h,m)} \mathcal{Q}_1$. Therefore, we have

$$\|\operatorname{grad} \hat{f}_{\pi(h,m)}(t\eta_{\pi(h,m)}) - \operatorname{grad} \hat{f}_{\pi(h,m)}(0)\|_{g_{(h,m)}} \le \frac{8}{7} \frac{15}{14d_*} \tilde{a}_L t \|\eta_{\pi(h,m)}\|_{g_{(h,m)}}, \quad \forall \ \ 0 \le t \le 1,$$

where $||h||_2 ||m||_2 \ge 14d_*/15$ from (B.1) is used. Note that both sides are independent of representation of $(h, m) \in \pi^{-1}(\pi(h, m))$, we completed the proof.