symmetry

MDPI

Article

An Intrinsic Scaled Riemannian Nonmonotone Conjugate Gradient Method on Stiefel Manifold

Yiyao Mei

School of Mathematics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; mmy@nuaa.edu.cn

Abstract

In this paper, we focus on the optimization problem on the Stiefel manifold. Based on the inverse process of the QR-type retraction, we propose a new intrinsic vector transport. In combination with Dai's nonmonotone conjugate gradient (CG) method, we present an intrinsic Riemannian nonmonotone CG method and its scaled version and establish the global convergence of the intrinsic scaled Riemannian nonmonotone CG method. Numerical results on a variety of optimization problems on Stiefel manifolds indicate the effectiveness of the proposed method.

Keywords: Riemannian optimization; Stiefel manifold; conjugate gradient method; retraction; vector transport

MSC: 49M37; 65K05; 90C30

1. Introduction

In this paper, we consider the following optimization problem

$$\min_{X \in St(n,p)} f(X) \tag{1}$$

where the feasible set, i.e.,

$$St(n,p) = \{X \in \mathbf{R}^{n\times p} \mid X^T X = I_p\},\ p \le n, \tag{2}$$

is called the Stiefel manifold, an $np - \frac{1}{2}p(p+1)$-dimensional submanifold in the vector space of $\mathbf{R}^{n\times p}$, and f is a continuously differentiable function defined on $St(n,p)$. The problem (1) arises from many scientific and engineering applications, for example, the symmetric eigenvalue problem [1,2], the trace ratio optimization problem [3], singular value decomposition [4,5], the orthogonal Procrustes problem [6,7], blind source separation in signal processing [8,9], orthogonal non-negative matrix factorization in document clustering and image processing [10,11], and the Kohn–Sham total energy minimization problem in electronic structure calculations [12,13]. For more applications of the optimization over the Stiefel manifold, we refer interested readers to [14–17] and the references therein. However, it is challenging to solve the problem (1) efficiently due to the nonconvexity of the orthogonality constraint (2).

The problem (1) has attracted much attention in the optimization community due to its wide range of applicationsand computational difficulty. A common approach for solving this problem is to use optimization methods [18] on the Euclidean space, where

Check for updates

Academic Editor: Hsien-Chung Wu

Received: 4 February 2026
Revised: 11 March 2026
Accepted: 14 March 2026
Published: 17 March 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

the iterative solution is mapped to the condition-restricted space at each step by some projection. This is useful but results in more computational costs because of the bad use of the solution space.

A popular approach for solving the problem (1) is taking this problem into the Riemannian manifold to use the recently developed Riemannian optimization technique. A comprehensive review of the Riemannian optimization technique is presented in [14–17]. Many classical optimization methods [19] in the Euclidean space have been generalized to forms on Riemannian manifolds. These methods are part of a vast literature on iterative methods for constrained problems, which includes projection-based methods for equilibrium and variational inequalities in general Banach spaces [20]. Generalized algorithms such as steepest descent methods [14], conjugate gradient methods [14,17,21–26], gradient projection methods [27], Barzilai–Borwein methods [27,28], Newton-like methods [14], quasi-Newton methods [21,29] and trust region methods [14,30] are often the methods of choice for efficiently solving the problem (1).

Our focus is on Riemannian conjugate gradient methods for solution of the minimization problem (1) on the Stiefel manifold, since these methods offer both theoretical and practical benefits. The theoretical benefit of these methods is that they generate sufficient descent search directions [31] and are globally convergent. The practical benefit of these methods is that they achieve efficient numerical performance, since only gradient information is needed, without the need to compute the Hessian matrix [14,17], and they are suitable for large-scale problems. Smith et al. [32] used an exponential map and parallel transport to generalize the optimization methods from Euclidean space to Riemannian manifolds. However, in some cases, using an exponential map or parallel translation is not computationally efficient, since there is no closed form for the parallel translation along the geodesic on the Stiefel manifold. Absil et al. [14] introduced a retraction that approximates the exponential map and maps the direction vector onto the manifold, as well as a vector transport that approximates parallel transport and maps a tangent vector on a point to another point on the manifold. QR decomposition and polar decomposition are usually used to construct the retraction on the Stiefel manifold. A commonly used way to construct the vector transport is the differentiated retraction operator. Ring and Wirth [21] proposed a Fletcher–Reeves Riemannian conjugate gradient method with retraction and vector transport and analyzed its global convergence under the assumption that the norm of a tangent vector does not increase after being transported. This assumption is usually referred to as the Ring–Wirth nonexpansive condition. However, not all vector transports satisfy this assumption. To remove this assumption from the global convergence analysis, Sato and Iwai [24] proposed a scaled vector transport and showed that the Fletcher–Reeves Riemannian conjugate gradient method with scaled vector transport generates a descent direction at every iteration and converges globally. Using the scaled vector transport, Sato [22], Sakai and Iiduka [31] developed a Dai–Yuan [33] Riemannian conjugate gradient method and a Hager–Zhang [34] Riemannian conjugate gradient method, respectively. However, this scaled vector transport violates the property of linearity. Zhu [25] derived two vector transports satisfying the Ring–Wirth nonexpansive condition and generalized Dai's nonmonotone conjugate gradient method [35] to the Riemannian case. Huang et al. [29] proposed an intrinsic representation of vector transport on the Stiefel manifold. This new vector transport is isometric and cheap but not traditional because it lacks a requirement of an associated retraction. By replacing vector transport with inverse retraction for search direction construction, Zhu and Sato [26] proposed and analyzed the Fletcher–Reeves and Dai–Yuan Riemannian conjugate gradient methods. Recently, Sato [23] presented a novel general framework that unifies existing Riemannian conjugate gradient methods.

Based on QR decomposition, we address the optimization problem on the Stiefel manifold. The main contributions of this paper are outlined as follows. (i) We propose a novel intrinsic vector transport via the inverse QR-type retraction, along with rigorous proofs of its existence and uniqueness. (ii) We develop an intrinsic Riemannian conjugate gradient algorithm by integrating this transport with Dai's nonmonotone conjugate gradient method [25,35], which is computationally efficient and strictly preserves the orthogonality constraint for large-scale problems. (iii) In order to guarantee the convergence of the intrinsic Riemannian conjugate gradient algorithm, by using the scaled vector transport, we present an intrinsic scaled Riemannian nonmonotone conjugate gradient method and establish its global convergence based on the Lipschitz continuity of the objective function.

This paper is organized as follows. In Section 2, we present notation and preliminaries on Riemannian conjugate gradient methods. In Section 3, based on the inverse process of the QR-type retraction, we propose an intrinsic vector transport, prove the existence and uniqueness of this new vector transport, and develop Dai's intrinsic Riemannian nonmonotone conjugate gradient method and its scaled version. In Section 4, we establish the global convergence of the intrinsic scaled Riemannian nonmonotone conjugate gradient method. Some numerical results are reported in Section 5, and conclusions are given in Section 6.

2. Notation and Preliminaries

Throughout this paper, we use the following notations. For a matrix ($A \in \mathbb{R}^{n\times p}$), A^T and $R(A)$ represent the transpose and the range of the matrix (A), respectively. $\mathcal{M}$ denotes a Riemannian manifold, and X is a point in $\mathcal{M}$. $T_X\mathcal{M}$ is the tangent space to $\mathcal{M}$ at point X, and $T\mathcal{M}$ denotes the tangent bundle of $\mathcal{M}$, i.e., $T\mathcal{M} = \bigcup_{X\in\mathcal{M}} T_X\mathcal{M}$. The set expressed as $T\mathcal{M} \oplus T\mathcal{M} := \{(\eta, \xi) \mid \eta, \xi \in T_X\mathcal{M}, X \in \mathcal{M}\}$ denotes the Whitney sum. $\langle\cdot,\cdot\rangle_X$ denotes the inner product on $T_X\mathcal{M}$, and $\|\xi\|_X = \sqrt{\langle\xi,\xi\rangle_X}$ represents the norm of a tangent vector ($\xi \in T_X\mathcal{M}$). ∇f denotes the gradient of function f.

The nonlinear conjugate gradient method for minimizing a smooth function (f) in Euclidean space uses the following update formula:

$$X_{k+1} = X_k + \alpha_k d_k, \tag{3}$$

where the step length (α_k) is determined by an exact or inexact line search and the search direction (d_k) is defined with different parameters, i.e., β_{k+1} [33,36–38], as follows:

$$d_{k+1} = -\nabla f(X_{k+1}) + \beta_{k+1} d_k \tag{4}$$

However, the update formula (3) and the search direction (4) are not directly applicable in Riemannian optimization, since the addition of two vectors is not defined on manifolds. One solution is to use the following retraction mapping and vector transport.

Definition 1 ([14]). *A retraction on a manifold ($\mathcal{M}$) is smooth mapping (R) from the tangent bundle ($T\mathcal{M}$) onto $\mathcal{M}$ with the following properties:*

1. *$R_X(0_X) = X$, where R_X denotes the restriction of R to $T_X\mathcal{M}$ at $X \in \mathcal{M}$ and 0_X is the zero element of $T_X\mathcal{M}$;*
2. *With the canonical identification of $T_{0_X}T_X\mathcal{M} \simeq T_X\mathcal{M}$, R_X satisfies $DR_X(0_X) = I_{T_X\mathcal{M}}$, where $DR_X(0_X)$ denotes the differential of R_X at 0_X and $I_{T_X\mathcal{M}}$ is the identity mapping on $T_X\mathcal{M}$.*

Definition 2 ([14]). *A vector transport ($\mathcal{T}$) on a manifold ($\mathcal{M}$) is a smooth mapping, i.e.,*

$$T\mathcal{M} \oplus T\mathcal{M} \to T\mathcal{M} : (\eta, \xi) \mapsto \mathcal{T}_\eta(\xi) \in T\mathcal{M},$$

satisfying the following properties for all $X \in \mathcal{M}$*:*

1. *(Associated retraction) There exists a retraction (R, called the retraction associated with* $\mathcal{T}$*) such that* $\mathcal{T}_\eta(\xi) \in T_{R_X(\eta)}\mathcal{M}$ *for all* $\eta, \xi \in T_X\mathcal{M}$*;*
2. *(Consistency)* $\mathcal{T}_{0_X}(\xi) = \xi$ *for all* $\xi \in T_X\mathcal{M}$*;*
3. *(Linearity)* $\mathcal{T}_\eta(a\xi + b\zeta) = a\mathcal{T}_\eta(\xi) + b\mathcal{T}_\eta(\zeta)$ *for all* $a, b \in \mathbb{R}$, $\eta, \xi, \zeta \in T_X\mathcal{M}$.

Definition 3 ([15]). *A vector transport* ($\mathcal{T}$) *is said to be isometric if it satisfies* $\langle \mathcal{T}_\eta(\xi), \mathcal{T}_\eta(\xi)\rangle_{R_X(\eta)} = \langle \xi, \xi\rangle_X$ *for all* $\eta, \xi \in T_X\mathcal{M}$*, where R is the retraction associated with* $\mathcal{T}$*.*

Let R be a retraction associated with a vector transport ($\mathcal{T}$). By using the retraction, the update Formula (3) may be generalized as

$$X_{k+1} = R_{X_k}(\alpha_k \eta_k), \tag{5}$$

where $\eta_k \in T_{X_k}\mathcal{M}$ is a tangent vector. It follows from (5) and Definition 2 that $\mathcal{T}_{\alpha_{k-1}\eta_{k-1}}(\eta_{k-1}) \in T_{X_k}\mathcal{M}$. Then, a Riemannian conjugate gradient direction is defined by the vector transport as follows:

$$\eta_k = \begin{cases} -grad f(X_k), & k = 0, \\ -grad f(X_k) + \beta_k \mathcal{T}_{\alpha_{k-1}\eta_{k-1}}(\eta_{k-1}), & k > 0, \end{cases} \tag{6}$$

where β_k is a Riemannian version of some parameter defined in Euclidean space.

Usually, we use the following differentiated retraction ($\mathcal{T}^R$) of R as a vector transport:

$$\mathcal{T}^R_\eta(\xi) = DR_X(\eta)[\xi] = \frac{d}{dt} R_X(\eta + t\xi)\,\big|_{t=0},\ X \in \mathcal{M}, \eta, \xi \in T_X\mathcal{M}.$$

Then, we have

$$\mathcal{T}^R_{\alpha_k\eta_k}(\eta_k) = DR_{X_k}(\alpha_k\eta_k)[\eta_k] = \frac{d}{dt} R_{X_k}(t\eta_k)\,\big|_{t=\alpha_k}.$$

In order to ensure that some minimum reduction in the objective value is obtained at each iteration, the step length (α_k) is determined to satisfy the following Wolfe conditions:

$$f(R_{X_k}(\alpha_k\eta_k)) \le f(X_k) + c_1\alpha_k\langle \nabla f(X_k), \eta_k\rangle_{X_k}, \tag{7}$$

$$|\langle \nabla f(R_{X_k}(\alpha_k\eta_k)), DR_{X_k}(\alpha_k\eta_k)[\eta_k]\rangle_{R_{X_k}(\alpha_k\eta_k)}| \le c_2|\langle \nabla f(X_k), \eta_k\rangle_{X_k}|, \tag{8}$$

or

$$\langle \nabla f(R_{X_k}(\alpha_k\eta_k)), DR_{X_k}(\alpha_k\eta_k)[\eta_k]\rangle_{R_{X_k}(\alpha_k\eta_k)} \ge c_2\langle \nabla f(X_k), \eta_k\rangle_{X_k}, \tag{9}$$

where $0 < c_1 < c_2 < 1$. In general, (7), (8) and (7), (9) are called the strong and weak Wolfe conditions, respectively. Regardless of whether strong or weak Wolfe conditions are used, the vector transport ($\mathcal{T}^R$) is not necessarily isometric, and it does not satisfy the following Ring–Wirth nonexpansive condition:

$$\langle \mathcal{T}^R_{\alpha_k\eta_k}(\eta_k), \mathcal{T}^R_{\alpha_k\eta_k}(\eta_k)\rangle_{R_{X_k}(\alpha_k\eta_k)} \le \langle \eta_k, \eta_k\rangle_{X_k}. \tag{10}$$

It follows from Definition 3 that condition (10) holds for every isometric vector transport. To establish global convergence for Riemannian conjugate gradient methods, we use the notion of scaled vector transport [24].

$$\mathcal{T}^S_{\alpha_k\eta_k}(\eta_k) = \begin{cases} \mathcal{T}^R_{\alpha_k\eta_k}(\eta_k), & if \|\mathcal{T}^R_{\alpha_k\eta_k}(\eta_k)\|_{R_{X_k}(\alpha_k\eta_k)} \leq \|\eta_k\|_{X_k}, \\ \frac{\|\eta_k\|_{X_k}}{\|\mathcal{T}^R_{\alpha_k\eta_k}(\eta_k)\|_{R_{X_k}(\alpha_k\eta_k)}}\mathcal{T}^R_{\alpha_k\eta_k}(\eta_k), & otherwise, \end{cases}$$

It is easy to verify that the Ring–Wirth nonexpansive condition (10) holds for the scaled vector transport ($\mathcal{T}^S$), but it is not linear. Recently, Huang et al. [29] proposed an intrinsic representation of vector transports for matrix manifolds including the Stiefel manifold. Their intrinsic vector transports are not only isometric but also cheap to compute, requiring $O(np^2)$ FLOPs.

3. An Intrinsic Scaled Riemannian Nonmonotone Conjugate Gradient Method

In this section, we propose a new vector transport based on the inverse process of the retraction via QR decomposition and present an intrinsic Riemannian conjugate gradient method in combination with Dai's nonmonotone conjugate method [35]. Based on the scaled vector transport, we develop an intrinsic scaled Riemannian nonmonotone conjugate gradient method.

The tangent space ($T_XSt(n,p)$) of the Stiefel manifold is expressed as

$$\begin{aligned} T_XSt(n,p) &= \{Z \mid X^TZ + Z^TX = 0, Z \in \mathbb{R}^{n\times p}\} \\ &= \{X\Omega + X_\perp K \mid \Omega^T = -\Omega \in \mathbb{R}^{p\times p}, K \in \mathbb{R}^{(n-p)\times p}\}, \end{aligned} \tag{11}$$

where $X_\perp$ is an $n \times (n-p)$ matrix whose columns form an orthonormal basis of the orthogonal complement of $R(X)$. The inner product of two matrices ($Y, Z \in T_XSt(n,p)$) is defined by $\langle Y, Z\rangle_X = tr(Y^TZ)$, which induces a norm expressed as $\|Z\|_X = \|Z\|_F$, i.e., the Frobenius norm of Z. Then, the projection of a point ($Y \in \mathbb{R}^{n\times p}$) onto $T_XSt(n,p)$ [14] is given by

$$P_X(Y) = Xskew(X^TY) + (I - XX^T)Y = Y - Xsym(X^TY), \tag{12}$$

where $skew(A) = (A - A^T)/2$ and $sym(A) = (A + A^T)/2$.

Lemma 1 ([14]). *Let $\bar{f}$ be a smooth function defined on a Riemannian manifold ($\mathcal{M}$), and let f denote the restriction of $\bar{f}$ to a Riemannian submanifold ($\bar{\mathcal{M}}$). The Riemannian gradient of f is equal to the projection of the gradient of $\bar{f}$ onto $T_x\bar{\mathcal{M}}$, i.e., $gradf(x) = P_x\nabla\bar{f}(x)$.*

It follows from Lemma 1 and (12) that the Riemannian gradient of the objective function ($f(X)$) is

$$gradf(X) = G - Xsym(X^TG), \tag{13}$$

where $G = \frac{\partial f(X)}{\partial X}$ is the gradient of $f(X)$ in Euclidean space.

The retractions on the Stiefel manifold have unique expression forms and excellent properties based on the constraint condition. Different choices of retractions lead to different performance outcomes.

Suppose that $A \in \mathbb{R}^{n\times p}$ has a rank of p; then, its QR decomposition ($A = QR$) is unique if we insist that the upper triangular matrix (R) has positive diagonal entries [1]. Hereafter, we always assume that the R factor of the QR factorization of a full rank matrix has positive diagonal elements. Let R^{qr} denote the retractions based on the QR decomposition. $Q_f(A)$ and $R_f(A)$ denote the Q and R factors of the QR decomposition of a matrix (A) with full column rank, respectively. The existing retraction via QR decomposition has the following form:

$$R^{qr}_X(Z) = Q_f(X + Z). \tag{14}$$

The differentiated retraction corresponding to (14) is

$$\begin{aligned}\mathcal{T}_Z^{qr}(Y) = DR_X^{qr}(Z)[Y] = R_X^{qr}(Z)skew\left[R_X^{qr}(Z)^T Y(R_f(X+Z))^{-1}\right] \\ + (I - R_X^{qr}(Z)R_X^{qr}(Z)^T)Y(R_f(X+Z))^{-1}.\end{aligned} \tag{15}$$

In contrast with the traditional vector transport given by the differential of the retraction, we propose an inverse approach that regards X_k as a retraction from X_{k+1}; then, the gradient direction derived by this retraction is in the same tangent space with $-\nabla f_{k+1}$.

Based on the inverse process of the retraction, we propose a new vector transport. Let $\mathcal{T}_{X_{k+1}}(X_k)$ be the new vector transport satisfying

$$X_k = R_{X_{k+1}}(\mathcal{T}_{X_{k+1}}(X_k)) \tag{16}$$

where $R_{X_{k+1}}$ is a retraction to $T_{X_{k+1}}St(n,p)$ at $X_{k+1} \in St(n,p)$. $\mathcal{T}_{X_{k+1}}(X_k)$ is the tangent vector at X_{k+1} that one would need to follow the retraction "backwards" to reach X_k. It follows from (16) that $\mathcal{T}_{X_{k+1}}(X_k)$ is in the same tangent space with $-\nabla f_{k+1}$ at X_{k+1}, which is also shown in Figure 1. Applying the new vector transport and the corresponding retraction by QR decomposition, the Riemannian conjugate gradient has the following form:

$$\begin{aligned}\eta_k &= \begin{cases} -gradf(X_k), & k=0, \\ -gradf(X_k) + \beta_k \mathcal{T}_{X_k}(X_{k-1}), & k>0, \end{cases} \\ X_{k+1} &= R_{X_k}^{qr}(\alpha_k \eta_k),\end{aligned} \tag{17}$$

where $R_{X_k}^{qr}$ is defined in (14) and $\mathcal{T}_{X_k}(X_{k-1})$ is given by (16).

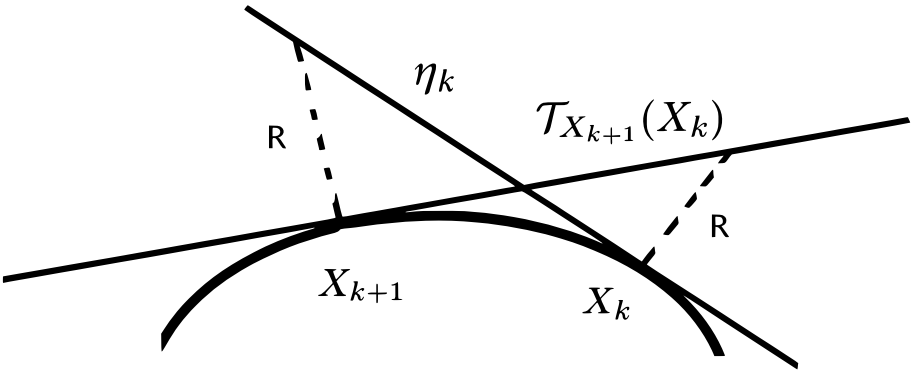


Figure 1. Sketch map of the new vector transport.

Now, we discuss the existence and uniqueness of the new vector transport.

Lemma 2. *If X_k and X_{k+1} are given by (17), then all leading principal submatrices of $X_k^T X_{k+1}$ are invertible.*

Proof. It follows from (17) that there is an upper triangular matrix ($\hat{R}$) with non-zero diagonal elements such that

$$X_{k+1}\hat{R} = X_k + \alpha_k \eta_k,$$

where η_k is a tangent vector at X_k. Then, we have

$$\begin{aligned}X_k^T X_{k+1}\hat{R} &= X_k^T X_k + \alpha_k X_k^T \eta_k \\ &= I + \alpha_k X_k^T \eta_k.\end{aligned}$$

According to (11), $\alpha_k X_k^T \eta_k$ is antisymmetric due to the fact that η_k lies in the tangent space at X_k^T. Hence, $I + \alpha_k X_k^T \eta_k$ and all its leading principal submatrices are invertible. By matrix partitioning, it is easy to verify that all leading principal submatrices of $X_k^T X_{k+1}$ are invertible. □

Let $S_k = I + X_{k+1}^T \mathcal{T}_{X_{k+1}}(X_k) = (S_{i,j}^{(k)}) \in \mathbb{R}^{p\times p}$. It is obvious that $X_{k+1}^T \mathcal{T}_{X_{k+1}}(X_k)$ is antisymmetric, since $\mathcal{T}_{X_{k+1}}(X_k)$ is in the tangent space at X_{k+1}. Therefore, all the diagonal elements of matrix S_k are 1.

Theorem 1. *If X_k and X_{k+1} are given by (17), then $\mathcal{T}_{X_{k+1}}(X_k)$ exists and is unique.*

Proof. From the definition (16) of $\mathcal{T}_{X_{k+1}}(X_k)$, we have

$$X_k \tilde{R} = X_{k+1} + \mathcal{T}_{X_{k+1}}(X_k),$$

where $\tilde{R} \in \mathbb{R}^{p\times p}$ is an upper triangular matrix. Then, we obtain

$$X_{k+1}^T X_k \tilde{R} = S_k. \tag{18}$$

Let $\tilde{X}_i$ and $\tilde{R}_i$ denote the $i \times i$ leading principal submatrices of $X_{k+1}^T X_k$ and $\tilde{R}$, respectively; e_i be the ith column of the $p \times p$ identity matrix; and $[x]_{1:i}$ be composed of the first i components of vector x. It follows from Lemma 2 and (18) that the upper triangular matrix ($\tilde{R}$) can be uniquely determined recursively as follows:

$$\begin{aligned} &for \quad i = 1, 2, \ldots, p \\ &\quad S_{i,i}^{(k)} = 1, \\ &\quad \tilde{X}_i [\tilde{R} e_i]_{1:i} \;=\; [S_k e_i]_{1:i}, \\ &\quad [S_k e_{i+1}]_{1:i}^T \;=\; -[e_{i+1}^T \tilde{X}_{i+1}]_{1:i} \tilde{R}_i. \end{aligned} \tag{19}$$

The system in (19) is solved recursively for the columns of $\tilde{R}$. This iteration is induced by the fact that S_k is skew-symmetric, with all diagonal entries equal to one. Therefore, the intrinsic vector transport can be uniquely obtained by

$$\mathcal{T}_{X_{k+1}}(X_k) = X_k \tilde{R} - X_{k+1}. \tag{20}$$

□

The intrinsic vector transport via QR decomposition is also useful on the Riemannian manifold.

Dai [35] proposed a nonlinear conjugate gradient method for unconstrained optimization in Euclidean space, and the β_{k+1} parameter is defined by

$$\beta_{k+1} = \frac{\|\nabla f(X_{k+1})\|^2}{\max\{y_k^T \eta_k, -\nabla f(X_k)^T \eta_k\}},$$

where $y_k = \nabla f(X_{k+1}) - \nabla f(X_k)$. The method is globally convergent, with a nonmonotone line-search condition instead of the Wolfe conditions. Zhu [25] developed a Dai Riemannian conjugate gradient method. The β_{k+1} parameter and the nonmonotone line-search condition are generalized to the Riemannian case as

$$\beta_{k+1}^D = \frac{\|gradf(X_{k+1})\|_{X_{k+1}}^2}{\max\{\langle gradf(X_{k+1}), \mathcal{T}_{X_{k+1}}(X_k)\rangle_{X_{k+1}} - \langle gradf(X_k), \eta_k\rangle_{X_k}, -\langle gradf(X_k), \eta_k\rangle_{X_k}\}} \tag{21}$$

and

$$f(R_{X_k}(\alpha_k\eta_k)) \leq \max\{f(X_k), \ldots, f(X_{k-\min\{m-1,k\}})\} + \delta\alpha_k\langle gradf(_k), \eta_k\rangle_{X_k}, \tag{22}$$

respectively, where m is some positive integer; $\delta, \lambda \in (0,\ 1)$; and $\alpha_k = \bar{\alpha}_k\lambda^m$, with $\bar{\alpha}_k$ being an initial guess at the k-th iteration. The Armijo-type condition (22) does not involve any vector transport when updating α_k, while the Wolfe conditions, i.e., (7)–(9), need to compute a vector transport.

Similar to Lemma 5 in [25], it can be shown that the following inequality holds for the intrinsic vector transport:

$$\langle gradf(X_k), \eta_k\rangle_{X_k} < 0. \tag{23}$$

It follows from (21) and (23) that $\beta^D_{k+1} > 0$ holds. Then, we choose $\beta_{k+1} \in [0, \beta^D_{k+1}]$.

An intrinsic Riemannian nonmonotone conjugate gradient algorithm is summarized as follows.

Remark 1. *The Lagrangian function of the problem* (1) *is*

$$\mathcal{L}(X, \Lambda) = f(X) - \frac{1}{2}tr(\Lambda(X^TX - I)),$$

where $\Lambda \in \mathbb{R}^{p\times p}$ *is a symmetric Lagrange multiplier matrix. It follows from* (13) *that the first-order optimality conditions, in the Euclidean sense, are* $X^TX = I$ *and* $\frac{\partial\mathcal{L}(X,\Lambda)}{\partial X} = G - XGX^T = 0$ *with* $\Lambda = G^TX$ *and* $G = \frac{\partial f(X)}{\partial X}$*, which is equivalent to* $gradf(X) = 0$ *in the Riemannian sense. Therefore,* $\|gradf(X_k)\|_{X_k} \leq \epsilon$ *for some constant* ($\epsilon > 0$) *is a reasonable stopping criterion for optimization on the Stiefel manifold.*

In view of the theoretical properties of the nonmonotone Armiji-type line search (22) and the numerical stability of Stiefel manifold optimization, the initial step-size guess ($\bar{\alpha}_{k+1}$) for the $(k+1)$-th iteration in Algorithm 1 is set to

$$\bar{\alpha}_{k+1} = \max\{\min\{\tilde{\alpha}_{k+1}, \alpha_{max}\}, \alpha_{min}\},$$

where $\tilde{\alpha}_{k+1} = \frac{tr(S_k^TS_k)}{|tr(Y_k^TS_k)|}$, with $S_k = \alpha_k\eta_k$ and $Y_k = gradf(X_{k+1}) - gradf(X_k)$.

Although our numerical experiments show that the intrinsic Riemannian nonmonotone conjugate gradient method is convergent, to ensure its global convergence, we adopt the following scaled vector transport:

$$\mathcal{T}^S_{X_{k+1}}(X_k) = \begin{cases} \mathcal{T}_{X_{k+1}}(X_k), & \|\mathcal{T}_{X_{k+1}}(X_k)\|_{X_{k+1}} \leq \|\eta_k\|_{X_k}, \\ \frac{\|\eta_k\|_{X_k}}{\|\mathcal{T}_{X_{k+1}}(X_k)\|_{X_{k+1}}}\mathcal{T}_{X_{k+1}}(X_k), & otherwise. \end{cases} \tag{24}$$

Then, the Ring0-Wirth nonexpansive condition (10) holds for the scaled vector transport ($\mathcal{T}^S$). By using the scaled vector transport, the β^D_{k+1} parameter is replaced by the following expression:

$$\beta^S_{k+1} = \frac{\|gradf(X_{k+1})\|^2_{X_{k+1}}}{max\{\langle gradf(X_{k+1}), \mathcal{T}^S_{X_{k+1}}(X_k)\rangle_{X_{k+1}} - \langle gradf(X_{X_k}), \eta_k\rangle_{X_k}, -\langle gradf(X_k), \eta_k\rangle_{X_k}\}}. \tag{25}$$

An intrinsic scaled Riemannian nonmonotone conjugate gradient algorithm is described as follows.

It is not difficult to show that the inequality (23) also holds for the intrinsic scaled vector transport.

Algorithm 1 IRNCG: Intrinsic Riemannian nonmonotone CG algorithm

Require: $\epsilon, \delta, \lambda \in (0,1), m \in \mathbf{N}^+$, $\alpha_{max} > \alpha_0 > \alpha_{min} > 0$, $X_0 \in St(n,p)$, $\eta_0 = -gradf(X_0)$, $k = 0$.

1: **while** $\|gradf(X_k)\|_{X_k} > \epsilon$ **do**
2: **if** $f(R_{X_k}(\alpha_k \eta_k)) \leq \max\{f(X_k), \cdots, f(X_{k-\min\{m-1,k\}})\} + \delta \alpha_k \langle gradf(X_k), \eta_k \rangle_{X_k}$ **then**
3: set $X_{k+1} = R_{X_k}(\alpha_k \eta_k)$, where R is defined by (14);
4: **else**
5: set $\alpha_k \leftarrow \lambda \alpha_k$ and go to Step 2;
6: **end if**
7: Compute $\eta_{k+1} = -gradf(X_{k+1}) + \beta^D_{k+1} \mathcal{T}_{X_{k+1}}(X_k)$, where

$$\beta^D_{k+1} = \frac{\|gradf(X_{k+1})\|^2_{X_{k+1}}}{max\{\langle gradf(X_{k+1}), \mathcal{T}_{X_{k+1}(X_k)} \rangle_{X_{k+1}} - \langle gradf(X_k), \eta_k \rangle_{X_k}, -\langle gradf(X_k), \eta_k \rangle_{X_k}\}},$$

and $\mathcal{T}_{X_{k+1}}(X_k)$ is given by (19) and (20);
8: Update $\alpha_{k+1} \in [\alpha_{min}, \alpha_{max}]$ and set $k := k+1$
9: **end while**

4. Global Convergence

In this section, we show the global convergence for the intrinsic scaled Riemannian nonmonotone conjugate gradient method. We assume that the objective function ($f(X)$) satisfies the following assumption.

Assumption 1. *The objective function ($f(X)$) is continuously differentiable on the Stiefel manifold ($St(n,p)$), and there exists a Lipschitzian constant ($L > 0$) such that*

$$|D(f \circ R_X)(t\eta)[\eta] - D(f \circ R_X)(0_X)[\eta]| \leq Lt$$

for all $\eta \in T_X St(n,p)$, $\|\eta\|_X = 1$, $X \in St(n,p)$, and $t \geq 0$.

For convenience, let

$$\nabla_k = \langle gradf(X_k), \eta_k \rangle_{X_k},$$

$$\tau_{k+1} = max\{\langle gradf(X_{k+1}), \mathcal{T}^S_{X_{k+1}}(X_k) \rangle_{X_{k+1}} - \nabla_k, -\nabla_k\}.$$

Lemma 3. *Suppose that f satisfies Assumption 1. Then, there exists a positive constant ($\mu > 0$) for Algorithm 2 such that*

$$\alpha_k = \min\{\bar{\alpha}_k, -\frac{\mu \nabla_k}{\|\eta_k\|^2_{X_k}}\} \text{ for all } k, \tag{26}$$

where $\bar{\alpha}_k$ is the initial guess at the k-th iteration. Furthermore, we have

$$\sum_{j \geq 1} \min_{i=1,\ldots,m} \{-\alpha_{mj+i-2} \nabla_{mj+i-2}\} < +\infty \tag{27}$$

Proof. The proof of Lemma 3 is similar to that of Lemma 6 in [25] and is omitted here. □

Now, we establish the global convergence result of Algorithm 2.

Theorem 2. *Suppose Assumption 1 holds and Algorithm 2 does not terminate in finitely many iterations. Then, the sequence ($\{X_k\}$) generated by Algorithm 2 converges in the form of*

$$\liminf_{k \to \infty} \|gradf(X_k)\|_{X_k} = 0. \tag{28}$$

Therefore, at least one accumulation point is a first-order critical point.

Algorithm 2 ISRNCG: Intrinsic scaled Riemannian nonmonotone CG algorithm

Require: $\epsilon, \delta, \lambda \in (0,1), m \in \mathbf{N}^+$, $\alpha_{max} > \alpha_0 > \alpha_{min} > 0$, $X_0 \in St(n,p)$, $\eta_0 = -gradf(X_0)$, $k = 0$.

1: **while** $\|gradf(X_k)\|_{X_k} > \epsilon$ **do**
2: **if** $f(R_{X_k}(\alpha_x \eta_k)) \leq \max\{f(X_k), \cdots, f(X_{k-\min\{m-1,k\}})\} + \delta\alpha_k \langle gradf(X_k), \eta_k\rangle_{X_k}$ **then**
3: set $X_{k+1} = R_{X_k}(\alpha_k \eta_k)$, where R is defined by (14);
4: **else**
5: set $\alpha_k \leftarrow \lambda\alpha_k$ and go to Step 2;
6: **end if**
7: Compute

$$\eta_{k+1} = -gradf(X_{k+1}) + \beta^S_{k+1} \mathcal{T}^S_{X_{k+1}}(X_k), \tag{29}$$

where $\mathcal{T}^S_{X_{k+1}}(X_k)$ and β^S_{k+1} are given by (24) and (25), respectively;
8: Update $\alpha_{k+1} \in [\alpha_{min}, \alpha_{max}]$ and set $k := k+1$;
9: **end while**

Proof. We prove Theorem 2 by contradiction. Suppose $\liminf_{k\to\infty} \|gradf(X_k)\|_{X_k} \neq 0$, which means that there exists a positive constant (γ) such that

$$\|gradf(X_k)\|_{X_k} \geq \gamma > 0 \text{ for all k.} \tag{30}$$

Using (25) and (29), we have

$$\nabla_{k+1} = \frac{\left\langle gradf(X_{k+1}), \mathcal{T}^S_{X_{k+1}}(X_k)\right\rangle_{X_{k+1}} - \tau_{k+1}}{\tau_{k+1}} \|gradf(X_{k+1})\|^2_{X_{k+1}},$$

and

$$\beta^S_{k+1} = \frac{\nabla_{k+1}}{\left\langle gradf(X_{k+1}), \mathcal{T}^S_{X_{k+1}}(X_k)\right\rangle_{X_{k+1}} - \tau_{k+1}}.$$

Using (23), we get

$$0 \leq \beta^S_{k+1} \leq \frac{\nabla_{k+1}}{\nabla_k}. \tag{31}$$

It follows from (29) and (31), as well as the fact that $\mathcal{T}^S$ satisfies the Ring–Wirth nonexpansive condition (10), that

$$\|\eta_{k+1}\|^2_{X_{k+1}} = -2\nabla_{k+1} - \|gradf(X_{k+1})\|^2_{X_{k+1}} + (\beta^S_{k+1})^2 \|\mathcal{T}^S_{X_{k+1}}(X_k)\|^2_{X_{k+1}},$$

and

$$\begin{aligned}
\frac{\|\eta_{k+1}\|^2_{X_{k+1}}}{\nabla^2_{k+1}} &\leq -\frac{2}{\nabla_{k+1}} - \frac{\|gradf(X_{k+1})\|^2_{X_{k+1}}}{\nabla^2_{k+1}} + \frac{\|\mathcal{T}^S_{X_{k+1}}(X_k)\|^2_{X_{k+1}}}{\nabla^2_k} \\
&\leq -\frac{2}{\nabla_{k+1}} - \frac{\|gradf(X_{k+1})\|^2_{X_{k+1}}}{\nabla^2_{k+1}} + \frac{\|\eta_k\|^2_{X_k}}{\nabla^2_k} \\
&= -\left(\frac{\|gradf(X_{k+1})\|_{X_{k+1}}}{\nabla_{k+1}} + \frac{1}{\|gradf(X_{k+1})\|_{X_{k+1}}}\right)^2 \\
&\quad + \frac{1}{\|gradf(X_{k+1})\|^2_{X_{k+1}}} + \frac{\|\eta_k\|^2_{X_k}}{\nabla^2_k} \\
&\leq \frac{1}{\|gradf(X_{k+1})\|^2_{X_{k+1}}} + \frac{\|\eta_k\|^2_{X_k}}{\nabla^2_k},
\end{aligned} \tag{32}$$

Combining (32) with the assumption of (30), we have

$$\begin{aligned}\frac{\|\eta_k\|_{X_k}^2}{\nabla_k^2} &\leq \sum_{i=1}^{k} \frac{1}{\|gradf(X_i)\|_{X_i}^2} + \frac{\|\eta_0\|_{X_0}^2}{grad_0^2} \\ &= \sum_{i=0}^{k} \frac{1}{\|gradf(X_i)\|_{X_i}^2} \leq \frac{k+1}{\gamma^2}.\end{aligned} \tag{33}$$

Then,

$$\frac{\nabla_k^2}{\|\eta_k\|_{X_k}^2} \geq \frac{\gamma^2}{k+1}. \tag{34}$$

On the other hand, from the second inequality in (32) and (33), we obtain

$$\frac{\|gradf(X_k)\|_{X_k}^2}{\nabla_k^2} + \frac{2}{\nabla_k} \leq \frac{\|\eta_{k-1}\|_{X_{k-1}}^2}{\nabla_{k-1}^2} \leq \frac{k}{\gamma^2},$$

which yields

$$\frac{k}{\gamma^2}\nabla_k^2 \quad \geq \quad 2\nabla_k + \|gradf(X_k)\|_{X_k}^2. \tag{35}$$

If

$$-\nabla_k \leq \frac{3}{8}\|gradf(X_k)\|_{X_k}^2,$$

then, from (35), we have

$$\nabla_k^2 \quad \geq \quad \frac{\gamma^2}{4k}\|gradf(X_k)\|_{X_k}^2.$$

Hence, from (23) and (30), we obtain

$$-\nabla_k \geq \min\{\frac{3\gamma^2}{8}, \frac{\gamma^2}{2\sqrt{k}}\}. \tag{36}$$

Using $\bar{\alpha}_k \geq \alpha_{min}$, (26), (34), and (36), we get

$$\begin{aligned}&\sum_{j\geq 1} \min_{i=1,\ldots,m} \{-\alpha_{mj+i-2}\nabla_{mj+i-2}\} \\ &\geq \sum_{j\geq 1} \min_{i=1,\ldots,m} \min\{-\alpha_{mj+i-2}\nabla_{mj+i-2}, \frac{\mu\nabla_{mj+i-2}}{\|\eta_{mj+i-2}\|_{X_{mj+i-2}}^2}\} \\ &\geq \sum_{j\geq 1} \min_{i=1,\ldots,m} \min\{\frac{3\gamma^2\alpha_{min}}{8}, \frac{\gamma^2\alpha_{min}}{2\sqrt{mj+i-2}}, \frac{\mu\gamma^2}{mj+i-1}\} \\ &\geq \sum_{j\geq 1} \min\{\frac{3\gamma^2\alpha_{min}}{8}, \frac{\gamma^2\alpha_{min}}{2\sqrt{m(j+1)}}, \frac{\mu\gamma^2}{m(j+1)}\} = +\infty,\end{aligned}$$

which contradicts (27). Therefore, (30) is impossible, and (28) follows.

The existence of a first-order critical accumulation point follows immediately from the compactness of the manifold in Assumption 1. □

5. Numerical Experiments

In this section, we demonstrate the effectiveness of Algorithms 1 and 2 on three kinds of application problems. All the numerical experiments are performed on a personal computer (2.5 GHz Quad-Core Intel Core i7 processor, 16 GB) by using MATLAB (Release 2016b). IRNCG and ISRNCG denote Algorithm 1 and Algorithm 2 respectively. CGQR and CGPOL denote the Riemannian conjugate gradient methods with the retraction via QR

decomposition [14] and polar decomposition [25], respectively. NPS denotes the Riemanian conjugate gradient methods with the RMIL+ conjugate gradient parameter [39].

In the actual implementation of the algorithms, we adopt the following stopping criteria:

$$\|gradf(X_k)\|_{X_k} \leq \epsilon,$$

or

$$\frac{\|X_k - X_{k-1}\|_F}{\sqrt{p}} \leq \epsilon_x \text{ and } \frac{|f(X_k) - f(X_{k-1})|}{|f(X_k)| + 1} \leq \epsilon_f. \tag{37}$$

The initial value of X_0 is taken from the first p columns of the identity matrix. The relevant parameters in Algorithms 1 and 2 are listed in Table 1.

Table 1. The input parameters for Algorithms 1 and 2.

Parameter	Value	Description
ϵ	10^{-6}	Tolerance for the norm of gradients
ϵ_x	10^{-6}	Tolerance for the difference of variables
ϵ_f	10^{-12}	Tolerance for the difference of function values
δ	10^{-4}	Nonmonotone line-search Armijo–Wolfe constant
λ	0.2	Step-length shrinkage factor
m	2	Nonmonotone line-search backward integers for consecutive previous function values
α_{max}	1	Upper threshold for step length
α_{min}	10^{-20}	Lower threshold for step length
α_0	10^{-3}	Initial step length

In the following tables, "fval" and "feasi" denote the objective function value and the feasibility ($\|X^TX - I\|_F$), respectively. "nrmg" and "rnrmg" represent the norm and relative norm of the gradient, respectively. "Iter" and "CPU" denote the number of iterations and the computing time (in seconds) for an algorithm to obtain an approximate solution of problem (1), respectively.

5.1. Symmetric Eigenvalue Problem

We consider the following symmetric eigenvalue problem [2,16]:

$$\min_{X \in St(n,p)} -tr(X^TAX), \tag{38}$$

where A is a symmetric $n \times n$ matrix. The corresponding objective function and its Euclidean gradient are

$$f(X) = -tr(X^TAX) \text{ and } G = \frac{\partial f(X)}{\partial X} = -2AX, respectively.$$

Let $p = 5$, $A = diag(1, 2, \ldots, n)$, and $A = M^TM$, where M is generated randomly by $randn(n)$ and the symmetric matrix (A) is taken from the the Harwell–Boeing sparse matrix collection [40]. The numerical results of applying the four algorithms to solve problem (38) are presented in Tables 2 and 3.

Table 2. Performance of four algorithms on the symmetric eigenvalue problems.

Algorithm	fval	feasi	nrmg	rnrmg	Iter	CPU
$A = diag(1, 2, \ldots, 1000)$						
IRNCG	−4990.0	7.2768×10^{-16}	0.0055	4.2881×10^{-6}	246	0.3872
ISRNCG	−4990.0	7.0873×10^{-16}	0.0064	5.0082×10^{-6}	245	0.3706
CGQR [14]	−4990.0	6.6674×10^{-16}	0.0114	8.8742×10^{-6}	253	0.4272
CGPOL [25]	−4990.0	4.5162×10^{-14}	0.0146	1.1377×10^{-5}	236	0.6973
NPS [39]	−4881.7	4.1491×10^{-16}	376.5	0.2956	12	0.0338
$A = diag(1, 2, \ldots, 5000)$						
IRNCG	−24,990.0	6.1457×10^{-16}	0.0464	7.1878×10^{-6}	513	43.86
ISRNCG	−24,990.0	6.1457×10^{-16}	0.0464	7.1878×10^{-6}	513	43.88
CGQR [14]	−24,990.0	8.1069×10^{-16}	0.0925	1.4308×10^{-5}	550	49.92
CGPOL [25]	−24,990.0	4.8662×10^{-13}	0.1456	2.2559×10^{-5}	520	102.7
NPS [39]	−24,987.0	9.5936×10^{-16}	41.88	0.0065	130	10.37
$A = diag(1, 2, \ldots, 10,000)$						
IRNCG	−49,990.0	7.8200×10^{-16}	0.2981	2.3079×10^{-5}	817	227.8
ISRNCG	−49,990.0	7.8200×10^{-16}	0.2981	2.3079×10^{-5}	817	229.0
CGQR [14]	−49,990.0	6.4091×10^{-16}	0.2923	2.2613×10^{-5}	657	200.0
CGPOL [25]	−49,990.0	5.7817×10^{-13}	0.2083	1.6139×10^{-5}	744	481.4
NPS [39]	−46,796.8	1.1610×10^{-15}	8819.8	0.6838	10	3.038
$A = diag(1, 2, \ldots, 20,000)$						
IRNCG	−99,990.0	8.1919×10^{-16}	7.3734	2.8569×10^{-4}	916	1020.6
ISRNCG	−99,990.0	8.1919×10^{-16}	7.3734	2.8569×10^{-4}	916	1009.5
CGQR [14]	−99,990.0	7.0497×10^{-16}	1.3301	5.1513×10^{-5}	870	1099.6
CGPOL [25]	−99,990.0	4.6927×10^{-11}	1.0277	3.9834×10^{-5}	930	2372.2
NPS [39]	−98,494.2	7.0573×10^{-16}	6567.6	0.2542	16	17.718
$A = M^T M$, $n = 1000$						
IRNCG	−19,429.0	5.8586×10^{-16}	0.0367	8.2039×10^{-6}	121	0.1900
ISRNCG	−19,429.0	5.8586×10^{-16}	0.0367	8.2039×10^{-6}	121	0.1874
CGQR [14]	−19,429.0	6.5337×10^{-16}	0.0390	8.7767×10^{-6}	119	0.1917
CGPOL [25]	−19,429.0	2.2121×10^{-14}	0.0170	3.8091×10^{-6}	121	0.3706
NPS [39]	−19,423.3	1.9979×10^{-16}	227.2	0.0511	38	0.0780
$A = M^T M$, $n = 5000$						
IRNCG	−98,985.0	8.9869×10^{-16}	0.0363	1.6213×10^{-6}	249	18.69
ISRNCG	−98,985.0	8.9869×10^{-16}	0.0363	1.6213×10^{-6}	249	18.81
CGQR [14]	−98,985.0	8.5529×10^{-16}	0.2901	1.2951×10^{-5}	234	17.87
CGPOL [25]	−98,985.0	6.9888×10^{-14}	0.2384	1.0682×10^{-5}	217	40.10
NPS [39]	−97,746.3	6.9211×10^{-16}	646.6	0.2893	20	1.641
$A = M^T M$, $n = 10,000$						
IRNCG	-1.9878×10^5	1.2192×10^{-15}	0.1075	2.4105×10^{-6}	177	39.45
ISRNCG	-1.9878×10^5	1.2192×10^{-15}	0.1075	2.4105×10^{-6}	177	39.17
CGQR [14]	-1.9878×10^5	7.4303×10^{-16}	0.3232	7.2473×10^{-6}	175	40.37
CGPOL [25]	-1.9878×10^5	2.4324×10^{-14}	0.1265	2.8376×10^{-6}	168	92.67
NPS [39]	-1.9860×10^5	3.9301×10^{-16}	3.8390	0.0859	46	10.77
$A = M^T M$, $n = 20,000$						
IRNCG	-3.9890×10^5	8.7341×10^{-16}	0.0871	9.7382×10^{-7}	206	184.7
ISRNCG	-3.9890×10^5	8.7341×10^{-16}	0.0871	9.7382×10^{-7}	206	182.0
CGQR [14]	-3.9890×10^5	8.7212×10^{-16}	0.1190	1.3301×10^{-6}	207	192.8
CGPOL [25]	-3.9890×10^5	5.7922×10^{-14}	0.7733	8.6410×10^{-5}	205	448.0
NPS [39]	-3.9416×10^5	9.4892×10^{-14}	5432.9	0.0608	69	63.95

Table 3. Linear eigenvalue problems from the Harwell–Boeing sparse matrix collection.

	IRNCG		ISRNCG		CGQR [14]		CGPOL [25]	
Matrix (n)	**Iter**	**CPU**	**Iter**	**CPU**	**Iter**	**CPU**	**Iter**	**CPU**
bcsstk08 (1074)	34	0.0661	34	0.0638	35	0.0705	42	0.1751
bcsstk11 (1473)	197	0.5815	197	0.5476	238	0.8119	282	1.7672
bcsstm11 (1473)	24	0.0884	24	0.0863	28	0.1003	33	0.2007
bcsstk13 (2003)	25	0.1753	25	0.1731	32	0.2078	25	0.3096
bcsstm24 (3562)	49	0.8544	49	0.8457	50	0.9236	54	1.8771
bcsstk15 (3948)	271	6.0112	271	5.9430	822	20.91	832	38.11
bcsstk18 (11,948)	105	34.03	105	34.26	106	36.54	126	95.71
bcsstk25 (15,439)	97	51.64	97	50.77	148	102.3	124	148.7
bcsstm25 (15,439)	37	18.96	37	18.65	55	29.77	56	66.98
e40r0000 (17,281)	505	324.5	714	458.7	650	464.2	387	612.8

From Table 2, it can be observed that all four algorithms find the same minimum, while ISRNCG is approximately the same as IRNCG in terms of the number of iterations and computing time and ISRNCG and IRNCG are more effective than CGQR and CGPOL in terms of the computing time in most cases. The unsatisfactory performance of NPS on this problem can be attributed to premature termination caused by the stopping criterion. The per iteration reduction provided by the stopping rule falls below a predefined threshold, which prevents the algorithm from achieving its intended convergence behavior.This observation also provides evidence for the robustness of the proposed algorithm. It can also be observed from Table 3 that ISRNCG and IRNCG are the same in terms of the number of iterations; that ISRNCG has a slightly shorter computing time than IRNCG, except for matrix e40r0000; and that ISRNCG and IRNCG are superior to CGQR and CGPOL in terms of the number of iterations and the computing time.

5.2. Heterogeneous QuadraticMinimization Problem

We consider the following heterogeneous quadraticminimization problem [25,41]:

$$\min_{X \in St(n,p)} \sum_{i=1}^{p} x_i^T A_i x_i \tag{39}$$

where $A_i \in \mathbb{R}^{n \times n} (i = 1, 2, \ldots, p)$ are symmetric matrices and x_i is the ith column of the matrix ($X \in \mathbb{R}^{n \times p}$), i.e., $X = [x_1, x_2, \ldots, x_p]$. The corresponding objective function and Euclidean gradient are

$$f(X) = \sum_{i=1}^{p} x_i^T A_i x_i \text{ and } G = \frac{\partial f(X)}{\partial X} = 2[A_1 x_1, A_2 x_2, \ldots, A_p x_p], respectively.$$

The data matrices (A_i) are taken as follows:

$$A_i = diag\left(\frac{(i-1)n+1}{p} : \frac{1}{p} : \frac{n}{p}\right) \quad \text{(Case 1)}$$

and

$$A_i = diag\left(\frac{(i-1)n+1}{p} : \frac{1}{p} : \frac{n}{p}\right) + B_i + B_i^T, \quad \text{(Case 2)},$$

where $B_i = 0.1 \cdot randn(n)$.

Using four algorithms to solve problem (39), we obtain the numerical results shown in Table 4. It is observed from Table 4 that the stopping criterion (37) is achieved by all algorithms, while the orthogonality constraint is not satisfied by CGPOL and the gradient norm generated by CGPOL has not yet converged. Extensive experimental results show that CGPOL is inappropriate for this problem. IRNCG and ISRNCG are superior to CGQR in terms of computing time in most cases.

Table 4. Numerical results for the heterogeneous quadraticminimization problem.

Algorithm	fval	feasi	nrmg	rnrmg	Iter	CPU
Case 1, n = 5000, p = 5						
IRNCG	1.0003×10^4	3.0163×10^{-15}	0.0115	8.8141×10^{-6}	419	1.269
ISRNCG	1.0003×10^4	2.0734×10^{-15}	0.0126	9.6867×10^{-6}	499	1.398
CGQR [14]	1.0003×10^4	1.4587×10^{-14}	0.0129	9.9134×10^{-6}	470	1.466
CGPOL [25]	1.0176×10^4	0.0072	447.6	0.3475	15	0.4801
Case 1, n = 10,000, p = 10						
IRNCG	4.5006×10^4	6.1286×10^{-15}	0.0180	9.8780×10^{-6}	785	9.237
ISRNCG	4.5006×10^4	3.0831×10^{-15}	0.0182	9.9620×10^{-6}	809	10.36
CGQR [14]	4.5006×10^4	2.4239×10^{-14}	0.0851	4.6558×10^{-5}	650	9.969
CGPOL [25]	4.5466×10^4	0.0093	315.2	0.1729	12	6.101
Case 1, n = 20,000, p = 20						
IRNCG	1.9001×10^5	9.5235×10^{-15}	0.2343	9.0772×10^{-5}	518	7.966
ISRNCG	1.9001×10^5	5.5952×10^{-15}	0.0205	7.9243×10^{-6}	797	12.14
CGQR [14]	1.9001×10^5	5.4019×10^{-14}	0.2137	8.2798×10^{-5}	918	16.84
CGPOL [25]	1.8067×10^5	0.9214	2.539×10^4	9.8357	14	0.3066
Case 2, n = 1000, p = 10						
IRNCG	4489.4	2.8693×10^{-15}	0.0018	9.9104×10^{-6}	839	17.64
ISRNCG	4489.4	2.9632×10^{-15}	0.0018	9.9071×10^{-6}	972	20.22
CGQR [14]	4489.9	3.8161×10^{-15}	0.6692	0.0036	1001	22.71
CGPOL [25]	4553.0	0.0260	80.1669	0.4309	14	0.4204
Case 2, n = 2500, p = 25						
IRNCG	2.9929×10^4	8.3101×10^{-15}	0.0241	8.1487×10^{-4}	1001	249.3
ISRNCG	2.9929×10^4	7.0894×10^{-15}	0.0605	2.0498×10^{-4}	919	251.2
CGQR [14]	2.9931×10^4	8.5606×10^{-15}	0.9225	0.0031	1001	269.1
CGPOL [25]	3.0134×10^4	0.0141	164.52	0.5563	11	5.023
Case 2, n = 5000, p = 50						
IRNCG	1.2223×10^5	2.0528×10^{-14}	0.0762	1.7700×10^{-4}	1001	6684.5
ISRNCG	1.2223×10^5	2.2361×10^{-15}	0.3182	7.3932×10^{-4}	1001	6768.8
CGQR [14]	1.2224×10^5	1.7672×10^{-14}	1.6185	0.0038	1001	7576.9
CGPOL [25]	1.2220×10^5	1.4393	1.2472×10^4	28.9748	10	135.5

5.3. Orthogonal Procrustes Problem

We consider the following orthogonal Procrustes problem [6,7]:

$$\min_{X \in St(n,p)} \frac{1}{2}||AX - B||_F^2, \tag{40}$$

where $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times p}$. The corresponding objective function and Euclidean gradient are

$$f(X) = \frac{1}{2}\|AX - B\|_F^2 \text{ and } G = A^T(AX - B), \textit{respectively}.$$

Data matrices A and B are set to $A = U\Sigma V^T$ and $B = AQ$, respectively, where $U, V \in \mathbb{R}^{n\times n}$ are random orthogonal matrices, Σ is a diagonal matrix with positive diagonal elements ($\sigma_i (i = 1, 2, \cdots, n)$), and $Q \in \mathbb{R}^{n\times p}$ is a random orthonormal matrix. We considered two types of Σ. One is $\sigma_i (i = 1, 2, \cdots, n)$, generated randomly within the interval [10, 12], and the other is $\sigma_i = 1 + \frac{i}{100} (i = 1, 2, \cdots, n)$. The numerical results of using four algorithms to solve problem (40) are listed in Table 5.

Table 5. Numerical results for the Procrustes problem.

Algorithm	fval	feasi	nrmg	rnrmg	Iter	CPU
$\sigma_i \in$ [10,12] , n = 500, p = 100						
IRNCG	1.0378×10^{-14}	4.6705×10^{-15}	7.2979×10^{-7}	2.3698×10^{-8}	38	0.6103
ISRNCG	1.0378×10^{-14}	4.6705×10^{-15}	7.2979×10^{-7}	2.3698×10^{-8}	38	0.5936
CGQR [14]	2.1632×10^{-14}	4.6478×10^{-15}	1.0655×10^{-6}	3.4566×10^{-8}	219	1.943
CGPOL [25]	5.8529×10^{-15}	3.3167×10^{-11}	8.7989×10^{-7}	2.8542×10^{-8}	45	0.9152
$\sigma_i \in$ [10, 12] , n = 1000, p = 200						
IRNCG	1.4776×10^{-14}	6.7904×10^{-15}	8.8964×10^{-7}	1.4363×10^{-8}	42	4.342
ISRNCG	1.4776×10^{-14}	6.7904×10^{-15}	8.8964×10^{-7}	1.4363×10^{-8}	42	4.263
CGQR [14]	2.6332×10^{-14}	6.8163×10^{-15}	1.1150×10^{-6}	1.7991×10^{-8}	290	17.00
CGPOL [25]	1.3801×10^{-14}	2.8355×10^{-8}	9.6940×10^{-7}	1.5644×10^{-8}	56	8.034
$\sigma_i = 1 + \frac{i}{100}$, n = 500, p = 100						
IRNCG	1.6835×10^{-13}	4.7177×10^{-15}	8.8084×10^{-7}	1.8515×10^{-7}	51	0.9267
ISRNCG	1.6835×10^{-13}	4.7177×10^{-15}	8.8084×10^{-7}	1.8515×10^{-7}	51	0.9137
CGQR [14]	3.6878×10^{-13}	4.6535×10^{-15}	1.1259×10^{-6}	2.3665×10^{-7}	472	4.669
CGPOL [25]	5.6644×10^{-14}	9.6325×10^{-12}	9.5715×10^{-7}	2.0115×10^{-7}	53	1.219
$\sigma_i = 1 + \frac{i}{100}$, n = 1000, p = 200						
IRNCG	2.6997×10^{-13}	6.8818×10^{-15}	1.0416×10^{-6}	3.4134×10^{-8}	118	12.25
ISRNCG	2.6997×10^{-13}	6.8818×10^{-15}	1.0416×10^{-6}	3.4134×10^{-8}	118	12.53
CGQR [14]	2.9019×10^{-10}	6.7220×10^{-15}	5.0006×10^{-5}	1.6387×10^{-6}	1001	54.29
CGPOL [25]	4.2362×10^{-14}	2.9436×10^{-8}	1.0050×10^{-6}	3.2929×10^{-8}	100	13.61

It is observed from Table 5 that these four algorithms are convergent and that IRNCG and ISRNCG are superior to CGQR and CGPOL in terms of the number of iterations and the computing time.

6. Conclusions

In this paper, we have presented an intrinsic vector transport based on the inverse process of the QR-type retraction without the traditional way via differentiated retraction and proposed an intrinsic Dai Riemannian nonmonotone CG method for the optimization on the Stiefel manifold. Based on the scaled vector transport, we have developed an intrinsic scaled Dai Riemannian nonmonotone CG method and demonstrated its global convergence. Numerical results show that the intrinsic scaled Dai Riemannian nonmonotone CG method is very promising. As a future work, we shall combine the idea of this paper with other CG methods to establish more efficient Riemannian CG methods.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available upon request from the author.

Acknowledgments: The author is very grateful to Hua Dai for his guidance and for revising the manuscript. The author is very grateful to the anonymous referees for their useful comments and suggestions, which greatly improved the representation of this paper.

Conflicts of Interest: The author declares no potential conflicts of interest.

References

1. Golub, G.H.; Van Loan, C.F. *Matrix Computations*, 4th ed.; The Johns Hopkins University Press: Baltimore, MD, USA, 2013.
2. Wen, Z.; Yang, C.; Liu, X.; Zhang, Y. Trace-penalty minimization for large-scale eigenspace computation. *J. Sci. Comput.* **2016**, *66*, 1175–1203. [CrossRef]
3. Ngo, T.T.; Bellalij, M.; Saad, Y. The trace ratio optimization problem. *SIAM Rev.* **2012**, *54*, 545–569. [CrossRef]
4. Liu, X.; Wen, Z.; Zhang, Y. Limited memory block Krylov subspace optimization for computing dominant singular value decompositions. *SIAM J. Sci. Comput.* **2013**, *35*, A1641–A1668. [CrossRef]
5. Sato, H.; Iwai, T. A Riemannian optimization approach to the matrix singular value decomposition. *SIAM J. Optim.* **2013**, *23*, 188–212. [CrossRef]
6. Bojanczyk, A.W.; Lutoborski, A. The procrustes problem for orthogonal Stiefel matrices. *SIAM J. Sci. Comput.* **1999**, *21*, 1291–1304. [CrossRef]
7. Gower, J.C.; Dijksterhuis, G.B. *Procrustes Problems*; Oxford University Press: Oxford, UK, 2004.
8. Wang, W.; Sanei, S.; Chambers, J.A. Penalty function-based joint diagonalization approach for convolutive blind separation of nonstationary sources. *IEEE Trans. Signal Process.* **2005**, *53*, 1654–1669. [CrossRef]
9. Chabriel, G.; Kleinsteuber, M.; Moreau, E.; Shen, H.; Tichavsky, P.; Yeredor, A. Joint matrices decompositions and blind source separation: A survey of methods, identification, and applications. *IEEE Signal Proc. Mag.* **2014**, *31*, 34–43. [CrossRef]
10. Pan, J.; Ng, M.K. Orthogonal nonnegative matrix factorization by sparsity and nuclear norm optimization. *SIAM J. Matrix Anal. Appl.* **2018**, *39*, 856–875. [CrossRef]
11. Chen, Y.-Y.; Xu, F.-F. Randomized algorithms for orthogonal nonnegative matrix factorization. *J. Oper. Res. Soc. China* **2023**, *11*, 327–345. [CrossRef]
12. Liu, X.; Wen, Z.; Wang, X.; Ulbrich, M.; Yuan, Y. On the analysis of the discretized Kohn-Sham density functional theory. *SIAM J. Numer. Anal.* **2015**, *53*, 1758–1785. [CrossRef]
13. Zhang, X.; Zhu, J.; Wen, Z.; Zhou, A. Gradient type optimization methods for electronic structure calculations. *SIAM J. Sci. Comput.* **2014**, *36*, C265–C289. [CrossRef]
14. Absil, P.A.; Mahony, R.; Sepulchre, R. *Optimization Algorithms on Matrix Manifolds*; Princeton University Press: Princeton, NJ, USA, 2009.
15. Huang W. Optimization Algorithms on Riemannian Manifolds with Applications. Ph.D. Thesis, The Florida State University, Tallahassee, FL, USA, 2013.
16. Hu, J.; Liu, X.; Wen, Z.; Yuan, Y.-X. A brief introduction to manifold optimization. *J. Oper. Res. Soc. China* **2020**, *8*, 199–248 [CrossRef]
17. Sato, H. *Riemannian Optimization and Its Applications*; Springer: Berlin/Heidelberg, Germany, 2021.
18. Nocedal, J.; Wright, S.J. *Numerical Optimization*; Springer: New York, NY, USA, 2006.
19. Wang Y.; Jiang M.; Shao H.; Wang X.; Wu T. Two globally convergent hybrid CG methods with adaptive restart strategy for Riemannian optimization. *Numer. Math. Theor. Meth. Appl.* **2025**, *18*, 1100–1124. [CrossRef]
20. Farid M.; Irfan S.S.; Khan M.F.; Khan S.A. Strong convergence of gradient projection method for generalized equilibrium problem in a Banach space. *J. Inequalities Appl.* **2017**, *2017*, 297. [CrossRef]
21. Ring, W.; Wirth, B. Optimization methods on Riemannian manifolds and their application to shape space. *SIAM J. Optim.* **2012**, *22*, 596–627. [CrossRef]
22. Sato, H. A Dai-Yuan-type Riemannian conjugate gradient method with the weak Wolfe conditions. *Comput. Optim. Appl.* **2016**, *64*, 101–118. [CrossRef]
23. Sato, H. Riemannian conjugate gradient methods: General framework and specific algorithms with convergence analyses. *SIAM J. Optim.* **2022**, *32*, 2690–2717. [CrossRef]
24. Sato, H.; Iwai, T. A new, globally convergent Riemannian conjugate gradient method. *Optimization* **2015**, *64*, 1011–1031. [CrossRef]
25. Zhu, X. A Riemannian conjugate gradient method for optimization on the Stiefel manifold. *Comput. Optim. Appl.* **2017**, *67*, 73–110. [CrossRef]
26. Zhu, X.; Sato, H. Riemannian conjugate gradient methods with inverse retraction. *Comput. Optim. Appl.* **2020**, *77*, 779–810. [CrossRef]
27. Oviedo, H.; Dalmau, O.; Lara, H. Two adaptive scaled gradient projection methods for Stiefel manifold constrained optimization. *Numer. Algorithms* **2021**, *87*, 1107–1127. [CrossRef]

28. Jiang, B.; Dai, Y.H. A framework of constraint preserving update schemes for optimization on Stiefel manifold. *Math. Program.* **2015**, *153*, 535–575. [CrossRef]
29. Huang, W.; Absil, P.A.; Gallivan, K.A. Intrinsic representation of tangent vectors and vector transports on matrix manifolds. *Numer. Math.* **2017**, *136*, 523–543. [CrossRef]
30. Absil, P.A.; Baker, C.G.; Gallivan, K.A. Trust-region methods on Riemannian manifolds. *Found. Comput. Math.* **2007**, *7*, 303–330.
31. Sakai, H.; Iiduka, H. Sufficient descent Riemannian conjugate gradient methods. *J. Optim. Theory Appl.* **2021**, *190*, 130–150. [CrossRef]
32. Smith, S.T. Optimization techniques on Riemannian manifolds. *Fields Inst. Commun.* **1994**, *3*, 113–135.
33. Dai, Y.H.; Yuan, Y. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* **1999**, *10*, 177–182. [CrossRef]
34. Hager, W.W.; Zhang, H. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J. Optim.* **2005**, *16*, 170–192. [CrossRef]
35. Dai, Y.H. A nonmonotone conjugate gradient algorithm for unconstrained optimization. *J. Syst. Sci. Complex.* **2002**, *15*, 139–145.
36. Fletcher, R.; Reeves, C.M. Function minimization by conjugate gradients. *Comput. J.* **1964**, *7*, 149–154. [CrossRef]
37. Hestenes, M.R.; Stiefel, E. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.* **1952**, *49*, 409–436. [CrossRef]
38. Polak, E.; Ribiére, G. Note sur la convergence de méthodes de directions conjuguées. *ESAIM Math. Model. Numer. Anal.* **1969**, *3*, 35–43. [CrossRef]
39. Salihu, N.; Kumam, P.; Salisu, S. Two efficient nonlinear conjugate gradient methods for Riemannian manifolds. *Comput. Appl. Math.* **2024**, *43*, 415. [CrossRef]
40. Duff, I.S.; Grimes, R.G.; Lewis, J.G. Sparse matrix test problems. *ACM Trans. Math. Softw.* **1989**, *15*, 1–14. [CrossRef]
41. Balogh, J.; Csendes, T.; Rapcsák, T. Some global optimization problems on Stiefel manifolds. *J. Glob. Optim.* **2004**, *30*, 91–101. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.