

RIEMANNIAN OPTIMIZATION ON TENSOR PRODUCTS OF GRASSMANN MANIFOLDS: APPLICATIONS TO GENERALIZED RAYLEIGH-QUOTIENTS

O. CURTEF^{†‡}, G. DIRR[†], AND U. HELMKE[†]

Abstract. We introduce a generalized Rayleigh-quotient ρ_A on the tensor product of Grassmannians $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ enabling a unified approach to well-known optimization tasks from different areas of numerical linear algebra, such as best low-rank approximations of tensors (data compression), geometric measures of entanglement (quantum computing) and subspace clustering (image processing). We briefly discuss the geometry of the constraint set $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$, we compute the Riemannian gradient of ρ_A , we characterize its critical points and prove that they are generically non-degenerated. Moreover, we derive an explicit necessary condition for the non-degeneracy of the Hessian. Finally, we present two intrinsic methods for optimizing ρ_A — a Newton-like and a conjugated gradient — and compare our algorithms tailored to the above-mentioned applications with established ones from the literature.

Key words. Riemannian optimization, Grassmann manifold, multilinear rank, best approximation of tensors, subspace clustering, entanglement measure, Newton method, conjugated gradient method.

AMS subject classifications. 14M15, 15A69, 65D19, 65F99, 65K10, 81P68

1. Introduction. The present paper addresses a constrained optimization problem, subsuming and extending optimization tasks which arise in various areas of applications such as (i) low-rank tensor approximation problems from signal processing and data compression, (ii) geometric measures of pure state entanglement from quantum computing, (iii) subspace reconstruction problems from image processing and (iv) combinatorial problems.

The problem can be stated as follows: Given a collection of integer pairs (m_j, n_j) with $1 \leq m_j \leq n_j$ for $j = 1, \dots, r$ and a Hermitian $N \times N$ matrix A with $N := n_1 n_2 \cdots n_r$, find the global maximizer of the trace function $\mathbf{P} \mapsto \text{tr}(A\mathbf{P})$. Here, \mathbf{P} is restricted to the set of all Hermitian projectors $\mathbf{P} : \mathbb{C}^N \rightarrow \mathbb{C}^N$ of rank $M := m_1 m_2 \cdots m_r$, which can be represented as a tensor product $\mathbf{P} := P_1 \otimes \cdots \otimes P_r$ of Hermitian projectors $P_j : \mathbb{C}^{n_j} \rightarrow \mathbb{C}^{n_j}$ of rank m_j . Thus, one is faced with the constrained optimization task

$$\max \text{tr}(A\mathbf{P}) \quad \text{subject to } \mathbf{P} \in \text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n}), \quad (1.1)$$

where $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ denotes the set of all Hermitian projectors of the above tensor type and (\mathbf{m}, \mathbf{n}) is a shortcut for $((m_1, n_1), \dots, (m_r, n_r))$. We will see that it makes sense to call the above objective function $\mathbf{P} \mapsto \text{tr}(A\mathbf{P}) =: \rho_A(\mathbf{P})$ the *generalized Rayleigh-quotient* of A with respect to the partitioning (\mathbf{m}, \mathbf{n}) .

To the best of the authors' knowledge, problem (1.1) has not been discussed in the literature in this general setting. However, depending on the structure of A as well as on the choice of (\mathbf{m}, \mathbf{n}) , problem (1.1) relates to well-known numerical linear algebra issues:

(i) For Hermitian matrices of rank-1, i.e. $A = vv^\dagger$, it reduces to a best low-rank approximation problem for the tensor $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_r}$ which satisfies $v = \text{vec}(\mathcal{A})$,

[†]Department of Mathematics, University of Würzburg, 97074 Würzburg, Germany

[‡]Corresponding author: curtef@mathematik.uni-wuerzburg.de

cf. [21, 28]. Classical application areas of such low-rank approximations can be found in statistics, signal processing and data compression [4, 20, 21, 31].

(ii) A recent application in quantum computing plays a central role in characterizing and quantifying pure state entanglement. Here, the distance of a pure state (tensor) to the set of all product states (rank-1 tensors) provides a geometric measure for entanglement [6, 23, 34].

(iii) Moreover, the challenging task of recovering subspaces of possibly different dimensions from noisy data — known as subspace detection or subspace clustering problem in computer vision and image processing [33] — can also be cast into the above setting. More precisely, for an appropriately chosen Hermitian matrix A the subspace clustering task can be characterized by problem (1.1) in the sense that for unperturbed data the global minima of the generalized Rayleigh-quotient are in unique correspondence with the sought subspaces. Numerical experiments in Section 4 support that even for noisy data the proposed optimization yields reliable approximations of the unperturbed subspaces.

(iv) In [3] a certain class of combinatorial problems are recast as optimization problems for trace functions on the special unitary group. For the case when A is a diagonal matrix, optimization task (1.1) is a generalization of the applications mentioned in [3].

Our solution to problem (1.1) is based on the fact that the constraint set $\text{Gr}^{\otimes}(\mathbf{m}, \mathbf{n})$ can be equipped with a Riemannian submanifold structure. This admits the use of techniques from Riemannian optimization — a rather new approach towards constrained optimization exploiting the geometrical structure of the constraint set in order to develop numerical algorithms [1, 14, 32]. In particular, we pursue two approaches: a Newton and a conjugated gradient method.

On a Riemannian manifold, the intrinsic Newton method is usually described by means of the Levi-Civita connection, performing iterations along geodesics, see [9, 29]. A more general approach via local coordinates was initiated by Shub in [27] and further discussed in [1, 13]. Here, we follow the ideas in [13] and use a pair of local parametrizations — normal coordinates for the push-forward and QR-type coordinates for the pull-back — satisfying an additional compatibility condition to preserve quadratic convergence. Thus we obtain an intrinsically defined version of the classical Newton algorithm with some computational flexibility. Nevertheless, for high-dimensional problems its iterations are expensive, both in terms of computational complexity and memory requirements. Therefore, we alternatively propose a conjugated gradient method, which has the advantage of algorithmic simplicity at a satisfactory convergence rate. In doing so, we suggest to replace the global line-search of the classical conjugated gradient method by a one-dimensional Newton-step, which yields a better convergence behavior near stationary points than the commonly used Armijo-rule.

As mentioned earlier, depending on the structure of A , the above-specified problems (i), (ii), (iii) and (iv) are particular cases of the optimization task (1.1). For the best low-rank approximation of a tensor the standard numerical approach is an alternating least-squares algorithm, known as higher-order orthogonal iteration (HOOI) [21]. Recently, several new methods also exploiting the geometric structure of the problem have been published. Newton algorithms have been proposed in [8, 18], quasi-Newton methods in [28], conjugated gradient and trust region methods in [17]. For high-dimensional tensors, all Riemannian Newton algorithms manifest similar problems: too high computational complexity and memory requirements. Our conju-

gated gradient method is however, a good candidate to solve large scale problems. It exhibits locally a good convergence behavior, comparable to that of the quasi-Newton methods in [28] at much lower computational costs, which considerably reduces the necessary CPU time.

For the problem of estimating a mixture of linear subspaces from sampled data points, cf. (iii), our numerical approach is an efficient alternative to the classical ones: *ad-hoc* type methods such as K-subspace algorithms [16], or probabilistic methods using a Maximum Likelihood framework for the estimation [30].

The paper is organized as follows. In Section 2, we familiarize the reader with the basic ingredients of Riemannian optimization. In particular, we address the following topics: the Riemannian submanifold structure of the constraint set $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$, its isometry to the r -fold cartesian product of Grassmannians, geodesics and parallel transport and the computation of the intrinsic gradient and Hessian for smooth objective functions. Section 3 is dedicated to the problem of optimizing the generalized Rayleigh-quotient ρ_A , including also a detailed discussion on its relation to problems (i), (ii), (iii) and (iv). Moreover, an analogy to the classical Rayleigh-quotient is also the subject of this section. We compute the gradient and the Hessian of the generalized Rayleigh-quotient and derive critical point conditions. We end the section with a result on the generic non-degeneracy of its critical points. In Section 4, a Newton-like and a conjugated gradient algorithm as well as numerical simulations tailored to the previously mentioned applications are given.

2. Preliminaries.

2.1. Riemannian structure of $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$. We start our study on the optimization task (1.1) with a brief summary on the necessary notations and basic concepts.

Let \mathfrak{her}_n be the set of all Hermitian $n \times n$ matrices A , i.e. $A \in \mathbb{C}^{n \times n}$ with $A^\dagger = A$, where A^\dagger refers to the conjugate transpose of A . Moreover, let SU_n be the Lie group of all special unitary matrices and \mathfrak{su}_n its Lie-algebra, i.e. $\Theta \in \text{SU}_n$ if and only if $\Theta^\dagger \Theta = I_n$, $\det \Theta = 1$ and, respectively, $\Omega \in \mathfrak{su}_n$ if and only if $\Omega^\dagger = -\Omega$ and $\text{tr}(\Omega) = 0$. The *Grassmannian*,

$$\text{Gr}_{m,n} := \{P \in \mathbb{C}^{n \times n} \mid P = P^\dagger = P^2, \text{tr}(P) = m\}, \quad (2.1)$$

is the set of all rank m Hermitian projection operators of \mathbb{C}^n . It is a smooth and compact submanifold of \mathfrak{her}_n with real dimension $2m(n-m)$, whose tangent space at P is given by

$$\text{T}_P \text{Gr}_{m,n} = \{[P, \Omega] := P\Omega - \Omega P \mid \Omega \in \mathfrak{su}_n\}, \quad (2.2)$$

cf. [13]. Hence, every element $P \in \text{Gr}_{m,n}$ and every tangent vector $\xi \in \text{T}_P \text{Gr}_{m,n}$ can be written as

$$P = \Theta \Pi_{m,n} \Theta^\dagger \quad \text{and} \quad \xi = \Theta \zeta_{m,n} \Theta^\dagger, \quad (2.3)$$

where $\Pi_{m,n}$ is the *standard projector* of rank m acting on \mathbb{C}^n and $\zeta_{m,n}$ denotes a tangent vector in the corresponding tangent space, i.e.

$$\Pi_{m,n} = \begin{bmatrix} I_m & 0 \\ 0 & 0 \end{bmatrix}, \quad \zeta_{m,n} = \begin{bmatrix} 0 & Z \\ Z^\dagger & 0 \end{bmatrix}, \quad Z \in \mathbb{C}^{m \times (n-m)}. \quad (2.4)$$

Whenever the values of m and n are clear from the context, we will use the shortcuts Π and ζ . With respect to the Riemannian metric induced by the Frobenius inner product of \mathfrak{her}_n , the Grassmannian $\text{Gr}_{m,n}$ is a Riemannian submanifold and the unique orthogonal projector onto $T_P \text{Gr}_{m,n}$ is given by

$$\text{ad}_P^2 X = [P, [P, X]], \quad X \in \mathfrak{her}_n. \quad (2.5)$$

We define the r -fold tensor product of Grassmannians Gr_{m_j, n_j} , $j = 1, \dots, r$ as the set

$$\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n}) := \{P_1 \otimes \dots \otimes P_r \mid P_j \in \text{Gr}_{m_j, n_j}, j = 1, \dots, r\} \quad (2.6)$$

of all rank- M Hermitian projectors of \mathbb{C}^N with $M := m_1 m_2 \dots m_r$ and $N := n_1 n_2 \dots n_r$, which can be represented as a Kronecker product $P_1 \otimes \dots \otimes P_r$. Here, (\mathbf{m}, \mathbf{n}) stands for the multi index

$$(\mathbf{m}, \mathbf{n}) := ((m_1, n_1), (m_2, n_2), \dots, (m_r, n_r)). \quad (2.7)$$

Then, $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ can be naturally equipped with a submanifold structure as the following result shows.

PROPOSITION 2.1. *The r -fold tensor product of Grassmannians $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ is a smooth and compact submanifold of \mathfrak{her}_N of real dimension $2 \sum_{i=1}^r m_i(n_i - m_i)$.*

Proof. We consider the following smooth action

$$\sigma : \text{SU}(\mathbf{n}) \times \mathfrak{her}_N \rightarrow \mathfrak{her}_N, \quad (\Theta, Y) \mapsto \Theta Y \Theta^\dagger,$$

of the compact Lie group

$$\text{SU}(\mathbf{n}) := \{\Theta := \Theta_1 \otimes \dots \otimes \Theta_r \mid \Theta_j \in \text{SU}_{n_j}\} \subset \text{SU}_N. \quad (2.8)$$

Let $X \in \mathfrak{her}_N$ be of the form $X := \Pi_1 \otimes \dots \otimes \Pi_r$, where Π_j denotes the standard projector in Gr_{m_j, n_j} . Then, the orbit $\mathcal{O}(X) := \{\Theta X \Theta^\dagger \mid \Theta \in \text{SU}(\mathbf{n})\}$ of X coincides with $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$. By [14] (pp. 44–46) we conclude that the r -fold tensor product of Grassmannians is a smooth and compact submanifold of \mathfrak{her}_N . Moreover, $\mathcal{O}(X) \cong \text{SU}(\mathbf{n})/\text{Stab}(X)$, where the stabilizer subgroup of X is given by

$$\begin{aligned} \text{Stab}(X) &:= \{\Theta \in \text{SU}(\mathbf{n}) \mid \Theta X \Theta^\dagger = X\} \\ &= \{\Theta \in \text{SU}(\mathbf{n}) \mid \Theta_j \Pi_j \Theta_j^\dagger = \Pi_j, j = 1, \dots, r\}. \end{aligned}$$

It follows easily that the dimension of $\text{Stab}(X)$ is $\sum_{i=1}^r [m_i^2 + (n_i - m_i)^2 - 1]$ and therefore,

$$\dim(\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})) = \dim \text{SU}(\mathbf{n}) - \dim \text{Stab}(X) = 2 \sum_{j=1}^r m_j(n_j - m_j)$$

is the dimension of the r -fold tensor product of Grassmannians. \square

REMARK 2.2. (a) Let $V \otimes W$ denote the tensor product of finite dimensional vector spaces V and W , cf. [12, 19] and let $X \otimes Y : V \otimes W \rightarrow V \otimes W$ be the tensor product of $X \in \text{End}(V)$ and $Y \in \text{End}(W)$, given by $v \otimes w \mapsto Xv \otimes Yw$, for all $v \in V$ and

$w \in W$. Moreover, let B_V and B_W be bases of V and W , respectively. Then, the matrix representation of $X \otimes Y$ with respect to the product basis $\{v \otimes w \mid v \in B_V, w \in B_W\}$ of $V \otimes W$ is given by the Kronecker product of the matrix representations of A and B with respect to B_V and B_W . This clarifies the relation between the “abstract” tensor product of linear maps and the Kronecker product of matrices and justifies the term “tensor product” of Grassmannians when we refer to $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$.

(b) It is a well-known fact that the Grassmannian $\text{Gr}_{m,n}$ is diffeomorphic to the Grassmann manifold $\text{Grass}_{m,n}$ of all m -dimensional subspaces of \mathbb{C}^n , cf.[14]. Therefore, $\text{Gr}_{m_1, n_1} \otimes \text{Gr}_{m_2, n_2}$ is diffeomorphic to

$$\{V_1 \otimes V_2 \mid V_1 \in \text{Grass}_{m_1, n_1}, V_2 \in \text{Grass}_{m_2, n_2}\} \subset \text{Grass}_{M, N}, \quad (2.9)$$

where $M := m_1 m_2$ and $N := n_1 n_2$.

Both items (a) and (b) readily generalize to an arbitrary number of Grassmannians.

We conclude this subsection by pointing out an isometry between the r -fold tensor product of Grassmannians $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ and the direct r -fold product of Grassmannians

$$\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) := \{(P_1, \dots, P_r) \mid P_j \in \text{Gr}_{m_j, n_j}, j = 1, \dots, r\}. \quad (2.10)$$

The vector spaces \mathfrak{her}_N and $\mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$ endowed with the inner products

$$\langle X, Y \rangle := \text{tr}(XY) \quad (2.11)$$

and

$$\langle (X_1, \dots, X_r), (Y_1, \dots, Y_r) \rangle := \text{tr}(X_1 Y_1) + \dots + \text{tr}(X_r Y_r), \quad (2.12)$$

induce a Riemannian submanifold structure on $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ and $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, respectively.

PROPOSITION 2.3. *The map*

$$\varphi : \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) \rightarrow \text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n}), \quad (P_1, \dots, P_r) \mapsto P_1 \otimes \dots \otimes P_r \quad (2.13)$$

is a diffeomorphism between $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$. Moreover, φ is a global Riemannian isometry when the right-hand side of (2.12) is replaced by

$$M_1 \text{tr}(X_1 Y_1) + \dots + M_r \text{tr}(X_r Y_r), \quad (2.14)$$

with $M_j := \prod_{k=1, k \neq j}^r m_k$, for $j = 1, \dots, r$.

Note that the isometry between $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ is very special, as in general the map

$$\mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r} \rightarrow \mathfrak{her}_N, \quad (X_1, \dots, X_r) \mapsto X_1 \otimes \dots \otimes X_r \quad (2.15)$$

fails even to be injective. For the proof of Proposition 2.3 we refer to the Appendix.

2.2. Geodesics and parallel transport. It is well-known that every Riemannian manifold \mathcal{M} carries a unique *Riemannian* or *Levi-Civita connection* ∇ , e.g. [1, 14, 32]. By means of ∇ , one defines parallel transport and geodesics as follows. Let $t \mapsto \mathcal{X}(t)$ be a vector field along a curve γ on \mathcal{M} , i.e. $\mathcal{X}(t) \in T_{\gamma(t)}\mathcal{M}$ for all $t \in \mathbb{R}$. Then, \mathcal{X} is defined to be *parallel along* γ if

$$\nabla_{\dot{\gamma}(t)}\mathcal{X}(t) = 0 \quad (2.16)$$

for all $t \in \mathbb{R}$. Given $\xi \in T_{\gamma(0)}\mathcal{M}$, there exists a unique parallel vector field \mathcal{X} along γ such that $\mathcal{X}(0) = \xi$ and the vector $\mathcal{X}(t) \in T_{\gamma(t)}\mathcal{M}$ is called the *parallel transport* of ξ to $T_{\gamma(t)}\mathcal{M}$ along γ . In particular, γ is called a *geodesic* on \mathcal{M} , if $\dot{\gamma}$ is parallel along γ .

For the Grassmann manifold $\text{Gr}_{m,n}$, the curve $t \mapsto \gamma(t) = e^{-t[\xi, P]}Pe^{t[\xi, P]}$ describes the geodesic through $P \in \text{Gr}_{m,n}$ in direction $\xi \in T_P\text{Gr}_{m,n}$, i.e. $\gamma(t)$ satisfies equation (2.16) with initial conditions $\gamma(0) = P$ and $\dot{\gamma}(0) = \xi$. Similarly, it can be verified that the parallel transport of $\eta \in T_P\text{Gr}_{m,n}$ to $T_{\gamma(t)}\text{Gr}_{m,n}$ along the geodesic through P in direction ξ is given by $\eta \mapsto e^{-t[\xi, P]}\eta e^{t[\xi, P]}$. These notions can be straight-forward generalized to the direct product of Grassmannians $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$.

2.3. The Riemannian gradient and Hessian. First, let us recall that the *Riemannian gradient* at $P \in \mathcal{M}$ of a smooth objective function $f : \mathcal{M} \rightarrow \mathbb{R}$ on a Riemannian manifold \mathcal{M} is defined as the unique tangent vector $\text{grad } f(P) \in T_P\mathcal{M}$ satisfying

$$df(P)(\xi) = \langle \text{grad } f(P), \xi \rangle \quad (2.17)$$

for all $\xi \in T_P\mathcal{M}$, where $df(P)$ denotes the differential (tangent map) of f at P . Moreover, if ∇ is the *Levi-Civita connection* on \mathcal{M} , then the *Riemannian Hessian* of f at P is the linear map $\mathbf{H}_f(P) : T_P\mathcal{M} \rightarrow T_P\mathcal{M}$ defined by

$$\mathbf{H}_f(P)\xi = \nabla_{\xi} \text{grad } f(P), \quad (2.18)$$

for all $\xi \in T_P\mathcal{M}$. Now, if \mathcal{M} is a submanifold of a vector space V , then (2.17) and (2.18) simplify as follows. Let \tilde{f} and $\tilde{\mathcal{X}}$ be smooth extensions of f and of the vector field $\text{grad } f$, respectively. Then,

$$\text{grad } f(P) = \pi_P(\nabla \tilde{f}(P)), \quad \mathbf{H}_f(P)\xi = \pi_P(D\tilde{\mathcal{X}}(P)\xi), \quad (2.19)$$

where π_P is the orthogonal projection onto $T_P\mathcal{M}$ and $\nabla \tilde{f}$ denotes the standard gradient of \tilde{f} on V .

For the generalized Rayleigh-quotient ρ_A on $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, explicit formulas of the gradient and Hessian will be given in Section 3.3.

3. The generalized Rayleigh-quotient. Let $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ be the r -fold tensor product of Grassmannians with (\mathbf{m}, \mathbf{n}) as in (2.7) and let $A \in \mathfrak{hcr}_N$, $N = n_1 n_2 \cdots n_r$. In the following, we analyze the constrained optimization problem

$$\max_{\mathbf{P} \in \text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})} \text{tr}(A\mathbf{P}), \quad (3.1)$$

which comprises problems from different areas, such as *multilinear low-rank approximations of a tensor*, *geometric measures of entanglement*, *subspace clustering* and *combinatorial optimization*. These applications are naturally stated on a tensor product space. However, for the special case of the Grassmann manifold they can be

reformulated on a direct product space. To this purpose, we define the *generalized Rayleigh-quotient* of the matrix A as

$$\rho_A : \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) \rightarrow \mathbb{R}, \quad \rho_A(P_1, \dots, P_r) := \text{tr} \left(A(P_1 \otimes \dots \otimes P_r) \right). \quad (3.2)$$

Based on the isometry between $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ and $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, we can rewrite problem (3.1) as an optimization task for ρ_A

$$\max_{(P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})} \rho_A(P_1, \dots, P_r). \quad (3.3)$$

In general this is not the case, as we have already pointed out in (2.15).

The term “generalized Rayleigh-quotient” is justified, since for $r = 1$ we obtain the classical Rayleigh-quotient $\rho_A(P) = \text{tr}(AP)$. In the sequel we want to point out some similarities and differences between the generalized and the classical Rayleigh-quotient. It is well known that under the assumption that there is a spectral gap between the eigenvalues of $A \in \mathfrak{her}_N$, there is a unique maximizer and a unique minimizer of the classical Rayleigh-quotient of A . Unfortunately, this is no longer the case for the generalized Rayleigh-quotient ρ_A . Global maximizers and global minimizers exist since the generalized Rayleigh-quotient is defined on a compact manifold, but unlike the classical case, it admits also local extrema as the following example shows. For the case when A is of rank one we refer to Example 3 in [21].

EXAMPLE 3.1. Let $A = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4) \in \mathfrak{her}_4$ be a diagonal matrix with $\lambda_2 > \lambda_3 > \lambda_4 > \lambda_1$ and $P_1^*, P_2^* \in \text{Gr}_{1,2}$ of the form

$$P_1^* = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad P_2^* = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \quad (3.4)$$

The maximum of ρ_A is obvious less or equal to λ_2 . Since $\rho_A(P_1^*, P_2^*) = \lambda_2$, we have (P_1^*, P_2^*) as the global maximizer of ρ_A . From (3.33) it follows that all $(P_1, P_2) \in \text{Gr}_{1,2} \times \text{Gr}_{1,2}$ with P_1 and P_2 diagonal, are critical points of ρ_A . In particular (P_2^*, P_1^*) is a critical point of ρ_A with $\rho_A(P_2^*, P_1^*) = \lambda_3 < \lambda_2$. Moreover, one can check by computing the Hessian of ρ_A at (P_2^*, P_1^*) , see (3.39), that (P_2^*, P_1^*) is actually a local maximizer of ρ_A . Comparative to the classical Rayleigh-quotient, this strange behavior results from the fact that not all 4×4 permutation matrices are of the form $\Theta_1 \otimes \Theta_2$, with $\Theta_1, \Theta_2 \in \text{SU}_2$.

While for the classical Rayleigh-quotient one knows that the maximizer and minimizer are orthogonal projectors onto the space spanned by the eigenvectors corresponding to the largest and, respectively, smallest eigenvalues of A , it is difficult to provide an analog characterization for the global extrema of the generalized Rayleigh-quotient for an arbitrary matrix A . But, for particular A and r such a characterization is possible.

(a) If $r = 2$ and A is of rank one, i.e. $A = \text{vec}(Y)\text{vec}(Y)^\dagger$, with $Y \in \mathbb{C}^{n_1 \times n_2}$, then the generalized Rayleigh-quotient can be rewritten as

$$\rho_A(P_1, P_2) = \text{tr}[\text{vec}(Y)\text{vec}(Y)^\dagger(P_1 \otimes P_2)] = \text{tr}(Y^\dagger P_1 Y P_2). \quad (3.5)$$

Under the assumption that Y has full rank and distinct singular values there exist one maximizer and one minimizer. The maximizer $(P_1^*, P_2^*) \in \text{Gr}^{\times 2}(\mathbf{m}, \mathbf{n})$ of ρ_A is given by the orthogonal projectors onto the space spanned by the $m_* := \min\{m_1, m_2\}$ left, respective right singular vectors corresponding to the largest m_* singular values. Similar for the minimizer, the singular vectors corresponding to the smallest m_* singular

values.

(b) If r is arbitrary and A diagonalizable via a transformation of $\mathrm{SU}(\mathbf{n}) = \{\Theta_1 \otimes \cdots \otimes \Theta_r \mid \Theta_j \in \mathrm{SU}_{n_j}\}$, then we can assume without loss of generality that A is diagonal. Moreover, if A can be written as $\Lambda_1 \otimes \cdots \otimes \Lambda_r$, with Λ_j diagonal, which is always the case when $A = A_1 \otimes \cdots \otimes A_r$, $A_j \in \mathfrak{her}_{n_j}$, then the generalized Rayleigh-quotient becomes a product of r decoupled classical Rayleigh-quotients. Hence, there is one maximizer and one minimizer. However, there is a dramatic change if A cannot be written as a Kronecker product of diagonal matrices. In this case ρ_A has also local extrema, as Example 3.1 shows. From (3.33) one can immediately formulate the following critical point characterization.

PROPOSITION 3.2. *Let $A \in \mathfrak{her}_N$ be diagonal. Then, $(P_1, \dots, P_r) \in \mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is a critical point of ρ_A if and only if P_j are permutations of the standard projectors Π_j , for all $j = 1, \dots, r$.*

3.1. Applications. There is a wide range of applications for problem (3.3) in areas such as signal processing, computer vision and quantum information. We briefly illustrate the broad potential of (3.3) by four examples.

3.1.1. Best multilinear rank- (m_1, \dots, m_r) tensor approximation. The problem of best approximation of a tensor by a tensor of lower rank is important in areas such as statistics, signal processing and pattern recognition. Unlike in the matrix case, there are several rank concepts for a higher order tensor, [21, 28]. For the scope of this paper, we focus on the *multilinear rank* case.

A finite dimensional complex tensor \mathcal{A} of order r is an element of a tensor product $V_1 \otimes \cdots \otimes V_r$, where V_1, \dots, V_r are complex vector spaces with $\dim V_j = n_j$. Such an element can have various representations, a common one is the description as an r -way array, i.e. after a choice of bases for V_1, \dots, V_r , the tensor \mathcal{A} is identified with $[a_{i_1 \dots i_r}]_{i_1=1, \dots, i_r=1}^{n_1, \dots, n_r} \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_r}$, see e.g. [28]. The j -th way of the array is referred to as the j -th *mode* of \mathcal{A} . A matrix $X \in \mathbb{C}^{q_j \times n_j}$ acts on a tensor $\mathcal{A} \in \mathbb{C}^{n_1 \times n_2 \times \cdots \times n_r}$ via *mode- j multiplication* \times_j , i.e.

$$(\mathcal{A} \times_j X)_{i_1 \dots i_{j-1} k_1 i_{j+1} \dots i_r} = \sum_{k_2=1}^{n_j} a_{i_1 \dots i_{j-1} k_2 i_{j+1} \dots i_r} x_{k_1 k_2}, \quad (3.6)$$

cf. [20, 28].

It is always possible to rearrange the elements of \mathcal{A} along one or, more general, several modes such that they form a matrix. Let l_1, \dots, l_q and c_1, \dots, c_p be ordered subsets of $1, \dots, r$ such that $\{l_1, \dots, l_q\} \cup \{c_1, \dots, c_p\} = \{1, \dots, r\}$. Moreover, consider the products $N_k := n_{l_{k+1}} \cdots n_{l_q}$, $N'_k := n_{c_{k+1}} \cdots n_{c_p}$, for $k = 0, \dots, q-1$ and $k = 0, \dots, p-1$, respectively. Then, the *matrix unfolding* of \mathcal{A} along (l_1, \dots, l_q) is a matrix $A_{(l_1, \dots, l_q)}$ of size $N_0 \times N'_0$ such that the element in position (i_1, \dots, i_r) of \mathcal{A} moves to position (s, t) in $A_{(l_1, \dots, l_q)}$, where

$$s := i_{l_q} + \sum_{k=1}^{q-1} (i_{l_k} - 1)N_k \quad \text{and} \quad t := i_{c_p} + \sum_{k=1}^{p-1} (i_{c_k} - 1)N'_k. \quad (3.7)$$

As an example, for a third order tensor $\mathcal{A} \in \mathbb{C}^{2 \times 2 \times 2}$ we obtain the following matrix

unfoldings as in [20]

$$A_{(1)} = \begin{bmatrix} a_{111} & a_{112} & a_{121} & a_{122} \\ a_{211} & a_{212} & a_{221} & a_{222} \end{bmatrix}, \quad A_{(2)} = \begin{bmatrix} a_{111} & a_{112} & a_{211} & a_{212} \\ a_{121} & a_{122} & a_{221} & a_{222} \end{bmatrix},$$

$$A_{(3)} = \begin{bmatrix} a_{111} & a_{121} & a_{211} & a_{221} \\ a_{112} & a_{122} & a_{212} & a_{222} \end{bmatrix}.$$

The *multilinear rank* of $\mathcal{A} \in \mathbb{C}^{n_1 \times \dots \times n_r}$ is the r -tuple (m_1, \dots, m_r) such that

$$m_1 = \text{rank } A_{(1)}, \quad \dots, \quad m_r = \text{rank } A_{(r)}. \quad (3.8)$$

To refer to the multilinear rank of \mathcal{A} we will use the notation $\text{rank}-(m_1, \dots, m_r)$ or $\text{rank } \mathcal{A} = (m_1, \dots, m_r)$. Given a tensor $\mathcal{A} \in \mathbb{C}^{n_1 \times \dots \times n_r}$, we are interested in finding the best $\text{rank}-(m_1, \dots, m_r)$ approximation of \mathcal{A} , i.e.

$$\min_{\text{rank}(\mathcal{B}) \leq (m_1, \dots, m_r)} \|\mathcal{A} - \mathcal{B}\|. \quad (3.9)$$

Here, $\|\mathcal{A}\|$ is the Frobenius norm of a tensor, i.e. $\|\mathcal{A}\|^2 = \langle \mathcal{A}, \mathcal{A} \rangle$ with

$$\langle \mathcal{A}, \mathcal{B} \rangle = \text{vec}(\mathcal{A})^\dagger \text{vec}(\mathcal{B}) = \sum_{i_1, \dots, i_r=1}^{n_1, \dots, n_r} \bar{a}_{i_1 \dots i_r} b_{i_1 \dots i_r}. \quad (3.10)$$

Here, $\text{vec}(\mathcal{A})$ refers to the matrix unfolding $A_{(1, \dots, r)} \in \mathbb{C}^{N \times 1}$. In the matrix case, the solution of the optimization problem (3.9) is given by a truncated SVD, cf. Eckart-Young theorem [7]. However, for the higher-order case, there is no equivalent of the Eckart-Young theorem. According to the Tucker decomposition [31] or the higher order singular value decomposition (HOSVD) [20], any $\text{rank}-(m_1, \dots, m_r)$ tensor can be written as a product of a *core* tensor \mathcal{S} and r Stiefel matrices $X_1 \in \mathbb{C}^{m_1 \times n_1}, \dots, X_r \in \mathbb{C}^{m_r \times n_r}$, i.e.

$$\mathcal{B} = \mathcal{S} \times_1 X_1 \times_2 \dots \times_r X_r, \quad X_j^\dagger X_j = I_{m_j}, \quad j = 1, \dots, r.$$

Thus, solving (3.9) is equivalent to solving the maximization problem

$$\max_{X_1, \dots, X_r} \|\mathcal{A} \times_1 X_1 \times_2 \dots \times_r X_r\|^2, \quad (3.11)$$

with $X_j^\dagger X_j = I_{m_j}$, $j = 1, \dots, r$, see e.g. [8]. Using vec -operation and Kronecker product language, one has

$$\text{vec}(\mathcal{A} \times_1 X_1 \times_2 \dots \times_r X_r) = \text{vec}(\mathcal{A})^\dagger (X_1 \otimes \dots \otimes X_r). \quad (3.12)$$

According to (3.10) and the properties of the *trace* function, the best multilinear $\text{rank}-(m_1, \dots, m_r)$ approximation problem becomes

$$\max_{(P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})} \text{tr} \left(A(P_1 \otimes \dots \otimes P_r) \right), \quad (3.13)$$

with $A = \text{vec}(\mathcal{A})\text{vec}(\mathcal{A})^\dagger$ and $P_j = X_j X_j^\dagger$, $j = 1, \dots, r$.

3.1.2. A geometric measure of entanglement. The task of characterizing and quantifying entanglement is a central theme in quantum information theory. There exist various ways to measure the difference between entangled and product states. Here, we discuss a geometric measure of entanglement, which is given by the Euclidean distance of $z \in \mathbb{C}^N$ with $\|z\| = 1$ to the set of all product states $\mathcal{P} = \{x_1 \otimes \cdots \otimes x_r \mid x_j \in \mathbb{C}^{n_j}, \|x_j\| = 1\}$, i.e.

$$\delta_E(z) := \min_{x \in \mathcal{P}} \|z - x\|^2. \quad (3.14)$$

Since any minimizer of δ_E is also a maximizer of

$$\max_{x_j \in \mathbb{C}^{n_j}, \|x_j\|=1} |z^\dagger(x_1 \otimes \cdots \otimes x_r)|, \quad (3.15)$$

and vice versa, computing the entanglement measure (3.14) is equivalent to solving

$$\max_{(P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})} \text{tr} \left(A(P_1 \otimes \cdots \otimes P_r) \right), \quad (3.16)$$

with $A = zz^\dagger$ and $P_1 = x_1 x_1^\dagger, \dots, P_r = x_r x_r^\dagger$. Note that (3.16) actually constitutes a best rank- $(1, \dots, 1)$ tensor approximation problem [6].

3.1.3. Subspace clustering. Subspace segmentation is a fundamental problem in many applications in computer vision (e.g. image segmentation) and image processing (e.g. image representation and compression). The problem of clustering data lying on multiple subspaces of different dimensions can be stated as follows:

Given a set of data points $X = \{x_l \in \mathbb{R}^n\}_{l=1}^L$ which lie approximately in $r \geq 1$ distinct subspaces S_k of dimension d_k , $1 \leq d_k < n$, identify the subspaces S_k without knowing in advance which points belong to which subspace.

Every d_k dimensional subspace $S_k \subset \mathbb{R}^n$ can be defined as the kernel of a rank $m_k = n - d_k$ orthogonal projector P_k of \mathbb{R}^{n_k} , with $n_k = n$ as

$$S_k = \{x \in \mathbb{R}^n \mid P_k x = 0\}. \quad (3.17)$$

Therefore, any point $x \in \bigcup_{k=1}^r S_k$ satisfies

$$\|P_1 x\| \cdot \|P_2 x\| \cdots \|P_r x\| = 0, \quad (3.18)$$

which is equivalent to

$$\text{tr}(xx^\top P_1) \text{tr}(xx^\top P_2) \cdots \text{tr}(xx^\top P_r) = \text{tr} \left((xx^\top \otimes \cdots \otimes xx^\top)(P_1 \otimes \cdots \otimes P_r) \right) = 0. \quad (3.19)$$

Thus, the problem of recovering the subspaces S_k from the data points X can be treated as the following optimization task:

$$\min_{P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})} \sum_{l=1}^L \prod_{k=1}^r \|P_k x_l\|^2 = \min_{P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})} \text{tr} \left(A(P_1 \otimes \cdots \otimes P_r) \right), \quad (3.20)$$

with $P := (P_1, \dots, P_r)$ and

$$A := \sum_{l=1}^L \underbrace{x_l x_l^\top \otimes \cdots \otimes x_l x_l^\top}_{r \text{ times}}. \quad (3.21)$$

We mention that here we have used the same notation $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ to refer to the direct r -fold product of *real* Grassmannians.

For best multilinear rank tensor approximation and subspace clustering applications, numerical experiments are presented at the end of Section 4.

3.1.4. A combinatorial problem. Let $\Lambda = (\lambda_{jk})_{j=1, k=1}^{n_2, n_1}$ be a given array of positive real numbers and let $m_1 \leq n_1$, $m_2 \leq n_2$ be fixed. Find m_1 columns and m_2 rows such that the sum of the corresponding entries λ_{jk} is maximal, i.e. solve the combinatorial maximization problem

$$\max_{\substack{J \subset \{1, \dots, n_2\} \\ |J|=m_2}} \max_{\substack{K \subset \{1, \dots, n_1\} \\ |K|=m_1}} \sum_{j \in J, k \in K} \lambda_{jk}. \quad (3.22)$$

We can permute m_1 columns and m_2 rows of Λ by right and left multiplication with permutations of the standard projectors Π_1 and Π_2 , respectively. Hence, problem (3.22) is solved by finding permutation matrices σ_1 and σ_2 which maximize:

$$\sum_{i,j} (\Pi_{\sigma_2} \Lambda \Pi_{\sigma_1})_{ij}, \quad (3.23)$$

where $\sum_{i,j}$ is the sum over all entries and $\Pi_{\sigma_1} := \sigma_1^\top \Pi_1 \sigma_1$, $\Pi_{\sigma_2} := \sigma_2^\top \Pi_2 \sigma_2$. The sum in (3.23) can be written as

$$\sum_{i,j} (\Pi_{\sigma_2} \Lambda \Pi_{\sigma_1})_{ij} = \sum_{i,j} \left((\Pi_{\sigma_1} \otimes \Pi_{\sigma_2}) \text{vec}(\Lambda) \right)_{ij} = \text{tr} \left(A (\Pi_{\sigma_1} \otimes \Pi_{\sigma_2}) \right), \quad (3.24)$$

where $A := \text{diag}(\text{vec}(\Lambda))$. The last equality in (3.24) holds since $\Pi_{\sigma_1} \otimes \Pi_{\sigma_2}$ is diagonal, too. According to Proposition 3.2, we have the following equivalence

$$\max_{\sigma_1, \sigma_2} \text{tr} \left(A (\Pi_{\sigma_1} \otimes \Pi_{\sigma_2}) \right) \equiv \max_{(P_1, P_2) \in \text{Gr}^{\times 2}(\mathbf{m}, \mathbf{n})} \text{tr} \left(A (P_1 \otimes P_2) \right). \quad (3.25)$$

Hence, we can embed the combinatorial maximization problem (3.22) into our continuous optimization task (3.3). The generalization of (3.22) to Λ being an arbitrary multi-array is straight-forward.

Problems of this type arise in multi-decision processes such as the following. Assume that a company has n_1 branches and each branch produces n_2 goods. If λ_{jk} denotes the gain of the j -th branch with the k -th good, then one could be interested to reduce the number of producers and goods to m_1 and m_2 , respectively, which give maximum benefit.

3.2. Riemannian optimization. We continue our investigation of problem (3.3) by computing the gradient and the Hessian of ρ_A . In the following lemma we establish multilinear maps $\Psi_{A,j}$, which will enable us to derive clear expressions for the gradient and the Hessian of ρ_A .

LEMMA 3.3. *Let $A \in \mathfrak{her}_N$ and $(X_1, \dots, X_r) \in \mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. Then, for all $j = 1, \dots, r$ there exists a unique map $\Psi_{A,j} : \mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r} \rightarrow \mathbb{C}^{n_j \times n_j}$ such that*

$$\text{tr} \left(A (X_1 \otimes \dots \otimes X_j Z \otimes \dots \otimes X_r) \right) = \text{tr} \left(\Psi_{A,j}(X_1, \dots, X_r) Z \right) \quad (3.26)$$

holds for all $Z \in \mathbb{C}^{n_j \times n_j}$. In particular, one has

$$\begin{aligned} \operatorname{tr} \left(A(X_1 \otimes \cdots \otimes X_r) \right) &= \operatorname{tr} \left(\Psi_{A,1}(I_{n_1}, X_2, \dots, X_r) X_1 \right) \\ &= \cdots = \operatorname{tr} \left(\Psi_{A,r}(X_1, \dots, X_{r-1}, I_{n_r}) X_r \right). \end{aligned} \quad (3.27)$$

Moreover, for $A := A_1 \otimes \cdots \otimes A_r$ the maps $\Psi_{A,j}$ exhibit the explicit form

$$\Psi_{A,j}(X_1, \dots, X_r) = \left(\prod_{k=1, k \neq j}^r \operatorname{tr}(A_k X_k) \right) A_j X_j. \quad (3.28)$$

Proof. Fix j and consider the linear functional

$$Z \mapsto \lambda_A(Z) := \operatorname{tr} \left(A(X_1 \otimes \cdots \otimes X_j Z \otimes \cdots \otimes X_r) \right).$$

By the Riesz representation theorem, there exists a unique $B_j \in \mathbb{C}^{n_j \times n_j}$ such that $\lambda_A(Z) = \operatorname{tr}(B_j Z)$ for all $Z \in \mathbb{C}^{n_j \times n_j}$. Therefore, the map $\Psi_{A,j}$ is given by $(X_1, \dots, X_r) \mapsto \Psi_{A,j}(X_1, \dots, X_r) := B_j$. It is straightforward to show that $\Psi_{A,j}$ is multilinear in X_1, \dots, X_r . Now, choosing $Z := X_j$ and $X_j := I_{n_j}$ in (3.26) immediately yields (3.27). Moreover, (3.28) follows from the trace equality

$$\operatorname{tr} (A_1 X_1 \otimes \cdots \otimes A_j X_j Z \otimes \cdots \otimes A_r X_r) = \left(\prod_{k=1, k \neq j}^r \operatorname{tr}(A_k X_k) \right) \operatorname{tr}(A_j X_j Z).$$

Thus the proof of Lemma 3.3 is complete. \square

REMARK 3.4. *The linear maps $\Psi_{A,j}$ constructed in the above proof are almost identical to the so-called partial trace operators — a well-known concept from multilinear algebra and quantum mechanics [2].*

Next, we show how to compute $\Psi_{A,j}(X_1, \dots, X_r)$ for given $(X_1, \dots, X_r) \in \mathfrak{her}_{n_1} \times \cdots \times \mathfrak{her}_{n_r}$ if A is not a pure tensor product $A_1 \otimes \cdots \otimes A_r$.

LEMMA 3.5. *Let $A \in \mathfrak{her}_N$ and $(X_1, \dots, X_r) \in \mathfrak{her}_{n_1} \times \cdots \times \mathfrak{her}_{n_r}$. Then, the (s, t) -position of $\Psi_{A,j}(X_1, \dots, X_r) \in \mathbb{C}^{n_j \times n_j}$ is given by*

$$\sum_{\substack{i_l=1, l \neq j \\ l=1, \dots, r}}^{n_l} e_{i_1}^\top \otimes \cdots \otimes e_s^\top \otimes \cdots \otimes e_{i_r}^\top A(X_1 \otimes \cdots \otimes X_r) e_{i_1} \otimes \cdots \otimes e_t \otimes \cdots \otimes e_{i_r}, \quad (3.29)$$

where $\{e_{i_l}\}_{i_l=1}^{n_l}$ denotes the standard basis of \mathbb{C}^{n_l} .

Proof. Let $1 \leq s, t \leq n_j$. Then, the element in the (s, t) position of the matrix $\Psi_{A,j}(X_1, \dots, X_r)$ is given by

$$\begin{aligned} e_s^\top \Psi_{A,j}(X_1, \dots, X_r) e_t &= \operatorname{tr} \left(\Psi_{A,j}(X_1, \dots, X_r) e_t e_s^\top \right) \\ &= \operatorname{tr} \left(A(X_1 \otimes \cdots \otimes X_j e_t e_s^\top \otimes \cdots \otimes X_r) \right) \\ &= \operatorname{tr} \left(A(X_1 \otimes \cdots \otimes X_r) (I_{n_1} \otimes \cdots \otimes e_t e_s^\top \otimes \cdots \otimes I_{n_r}) \right). \end{aligned}$$

Hence, (3.29) follows from the identity $I_{n_l} = \sum_{i_l=1}^{n_l} e_{i_l} e_{i_l}^\top$. \square

REMARK 3.6. Let $A \in \mathfrak{her}_N$ and $(X_1, \dots, X_r) \in \mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. A straightforward consequence of the identity

$$\mathrm{tr} \left(A(X_1 \otimes \dots \otimes Z \otimes \dots \otimes X_r) \right)^\dagger = \mathrm{tr} \left(A(X_1 \otimes \dots \otimes Z^\dagger \otimes \dots \otimes X_r) \right), \quad (3.30)$$

for all $Z \in \mathbb{C}^{n_j \times n_j}$, shows that $\Psi_{A,j}(X_1, \dots, I_{n_j}, \dots, X_r)$ is Hermitian. For simplicity of writing, whenever $(P_1, \dots, P_r) \in \mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is understood from the context, we use the following shortcut

$$\widehat{A}_j := \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_r). \quad (3.31)$$

Now, we can give an explicit formula for the Riemannian gradient of ρ_A and derive necessary and sufficient critical point conditions.

THEOREM 3.7. Let $A \in \mathfrak{her}_N$, $P := (P_1, \dots, P_r) \in \mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and let ρ_A be the generalized Rayleigh-quotient on $\mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$. Then, one has the following:

(i) The gradient of ρ_A at P with respect to the Riemannian metric (2.12) is

$$\mathrm{grad} \rho_A(P) = \left(\mathrm{ad}_{P_1}^2 \widehat{A}_1, \dots, \mathrm{ad}_{P_r}^2 \widehat{A}_r \right). \quad (3.32)$$

(ii) The critical points of ρ_A on $\mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ are characterized by

$$[P_j, \widehat{A}_j] = 0 \quad (3.33)$$

i.e. P_j , $j = 1, \dots, r$ is the orthogonal projector onto an m_j -dimensional invariant subspace of \widehat{A}_j .

Proof. (i) Fix $P := (P_1, \dots, P_r) \in \mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and let $\tilde{\rho}_A$ denote the canonical smooth extension of ρ_A to $\mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. Then,

$$D\tilde{\rho}_A(P)(X) = \sum_{j=1}^r \mathrm{tr} \left(A(P_1 \otimes \dots \otimes X_j \otimes \dots \otimes P_r) \right) = \sum_{j=1}^r \mathrm{tr}(\widehat{A}_j X_j), \quad (3.34)$$

for all $X := (X_1, \dots, X_r) \in \mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. From (2.12), we obtain that the gradient of $\tilde{\rho}_A$ at P is given by $\nabla \tilde{\rho}_A(P) = (\widehat{A}_1, \dots, \widehat{A}_r)$. Thus, according to (2.5) and (2.19),

$$\mathrm{grad} \rho_A(P) = \left(\mathrm{ad}_{P_1}^2 \widehat{A}_1, \dots, \mathrm{ad}_{P_r}^2 \widehat{A}_r \right). \quad (3.35)$$

(ii) $P := (P_1, \dots, P_r) \in \mathrm{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is a critical point of ρ_A iff $\mathrm{grad} \rho_A(P) = 0$. This is equivalent to

$$P_j[P_j, \widehat{A}_j] = [P_j, \widehat{A}_j]P_j, \quad (3.36)$$

for all $j = 1, \dots, r$. By multiplying (3.36) once from the left with P_j and once from the right with P_j , we obtain that $P_j \widehat{A}_j = P_j \widehat{A}_j P_j$ and $\widehat{A}_j P_j = P_j \widehat{A}_j P_j$. Hence, the conclusion $[P_j, \widehat{A}_j] = 0$ holds for all $j = 1, \dots, r$. \square

As a consequence of Theorem 3.7, we immediately obtain the following necessary and sufficient critical point condition.

COROLLARY 3.8. *Let $A \in \mathfrak{her}_N$, $P := (P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and let $\Theta_j \in \text{SU}_{n_j}$ be such that $\Theta_j^\dagger P_j \Theta_j = \Pi_j$, where Π_j is the standard projector in Gr_{m_j, n_j} . We write*

$$\Theta_j^\dagger \hat{A}_j \Theta_j = \begin{bmatrix} \Psi'_j & \Psi'''_j \\ \Psi_j^{\prime\prime\dagger} & \Psi''_j \end{bmatrix}, \quad (3.37)$$

with $\Psi'_j \in \mathfrak{her}_{m_j}$, $\Psi''_j \in \mathfrak{her}_{n_j - m_j}$, and $\Psi'''_j \in \mathbb{C}^{m_j \times (n_j - m_j)}$. Then P is a critical point of ρ_A if and only if

$$\Psi'''_j = 0, \quad (3.38)$$

for all $j = 1, \dots, r$.

For the rest of this section we are concerned with the computation of the Riemannian Hessian of ρ_A and also with its non-degeneracy at critical points.

THEOREM 3.9. *Let $A \in \mathfrak{her}_N$ and $P := (P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$. Then, the Riemannian Hessian of ρ_A at P is the unique self-adjoint operator*

$$\begin{aligned} \mathbf{H}_{\rho_A}(P) : \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) &\rightarrow \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}), \\ \xi := (\xi_1, \dots, \xi_r) &\mapsto \mathbf{H}_{\rho_A}(P)(\xi) := \left(\mathbf{H}_1(\xi), \dots, \mathbf{H}_r(\xi) \right), \end{aligned} \quad (3.39)$$

defined by

$$\mathbf{H}_j(\xi) := -\text{ad}_{P_j} \text{ad}_{\hat{A}_j} \xi_j + \sum_{k=1, k \neq j}^r \text{ad}_{P_j}^2 \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, \xi_k, \dots, P_r), \quad (3.40)$$

where \hat{A}_j is the shortcut for $\Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_r)$.

Proof. Let $(\tilde{\mathcal{X}}_1, \dots, \tilde{\mathcal{X}}_r)$ denote a smooth extension of $\text{grad } \rho_A$ to $\mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. According to (3.32), we can choose $P \mapsto \tilde{\mathcal{X}}_j(P) = \text{ad}_{P_j}^2 \hat{A}_j$. Then,

$$\begin{aligned} D\tilde{\mathcal{X}}_j(P)(X) &= \text{ad}_{X_j} \text{ad}_{P_j} \hat{A}_j + \text{ad}_{P_j} \text{ad}_{X_j} \hat{A}_j \\ &+ \sum_{k=1, k \neq j}^r \text{ad}_{P_j}^2 \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, X_k, \dots, P_r), \end{aligned} \quad (3.41)$$

for all $P := (P_1, \dots, P_r)$ and $X := (X_1, \dots, X_r)$ in $\mathfrak{her}_{n_1} \times \dots \times \mathfrak{her}_{n_r}$. Notice that, the derivative of the linear map $P_k \mapsto \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_k, \dots, P_r)$ in direction $X_k \in \mathfrak{her}_{n_k}$ ($k \neq j$) is $\Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, X_k, \dots, P_r)$. Applying (2.5) and (2.19), the Riemannian Hessian of ρ_A at $P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is given by (3.39) and (3.40). Here, we have used the following two facts:

(i) Clearly, $\text{ad}_{\hat{A}_j} \xi_j$ is skew-hermitian and hence

$$-\text{ad}_{P_j} \text{ad}_{\hat{A}_j} \xi_j = \text{ad}_{P_j} \text{ad}_{\xi_j} \hat{A}_j \quad (3.42)$$

is in the tangent space $\text{T}_{P_j} \text{Gr}_{m_j, n_j}$ for all $\xi_j \in \text{T}_{P_j} \text{Gr}_{m_j, n_j}$.

(ii) A straightforward computation shows that $\text{ad}_{\xi_j} \text{ad}_{P_j} \hat{A}_j$ is in the orthogonal complement of $\text{T}_{P_j} \text{Gr}_{m_j, n_j}$ and hence

$$\text{ad}_{P_j}^2 \text{ad}_{\xi_j} \text{ad}_{P_j} \hat{A}_j = 0 \quad (3.43)$$

for all $\xi_j \in T_{P_j} \text{Gr}_{m_j, n_j}$. \square

By restricting the tangent vectors $(\xi_1, \dots, \xi_r) \in T_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ to the vectors of the form $(0, \dots, \xi_j, \dots, 0)$, it follows immediately a necessary condition for the non-degeneracy of the Hessian at local extrema.

THEOREM 3.10. *Let $A \in \mathfrak{hcr}_N$, and $P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ be a local maximizer (local minimizer) of ρ_A . If $\mathbf{H}_{\rho_A}(P)$ is non-degenerate, then for all $j = 1, \dots, r$ the equality*

$$\sigma(\Psi'_j) \cap \sigma(\Psi''_j) = \emptyset, \quad (3.44)$$

holds with Ψ'_j and Ψ''_j as in (3.37). Here, $\sigma(X)$ denotes the spectrum of X .

REMARK 3.11. *In the case when $A \in \mathfrak{hcr}_N$ can be diagonalized by elements in $\text{SU}(\mathbf{n}) = \{\Theta_1 \otimes \dots \otimes \Theta_r \mid \Theta_j \in \text{SU}_{n_j}\}$, condition (3.44) is also sufficient for the nondegeneracy of the Hessian of ρ_A at local extrema.*

In the remaining part of the section we derive a genericity statement concerning the critical points of the generalized Rayleigh-quotient. The result is a straightforward consequence of the parametric transversality theorem [15]. Let $V, \mathcal{M}, \mathcal{N}$ be smooth manifolds and $F : V \times \mathcal{M} \rightarrow \mathcal{N}$ a smooth map. Moreover, let $T_{(A,P)}F : V \times T_P \mathcal{M} \rightarrow T_{F(A,P)} \mathcal{N}$ denote the tangent map of F at $(A, P) \in V \times \mathcal{M}$. We say that F is *transversal* to a submanifold $S \subset \mathcal{N}$ and write $F \pitchfork S$ if

$$\text{Im } T_{(A,P)}F + T_{F(A,P)}S = T_{F(A,P)}\mathcal{N}, \quad (3.45)$$

for all $(A, P) \in F^{-1}(S)$. Then, the parametric transversality theorem states the following.

THEOREM 3.12. *([15]) Let $V, \mathcal{M}, \mathcal{N}$ be smooth manifolds and S a closed submanifold of \mathcal{N} . Let $F : V \times \mathcal{M} \rightarrow \mathcal{N}$ be a smooth map, let $A \in V$ and define $F_A : \mathcal{M} \rightarrow \mathcal{N}$, $F_A(P) := F(A, P)$. If $F \pitchfork S$, then the set*

$$\{A \in V \mid F_A \pitchfork S\} \quad (3.46)$$

is open and dense.

Now, let $f_A : \mathcal{M} \rightarrow \mathbb{R}$ be a smooth function depending on a parameter $A \in V$ and consider the map

$$F : V \times \mathcal{M} \rightarrow T^*\mathcal{M}, \quad F(A, P) := df_A(P), \quad (3.47)$$

where $T^*\mathcal{M}$ is the cotangent bundle of \mathcal{M} and $df_A(P)$ denotes the differential of f_A at $P \in \mathcal{M}$. With these notations, our genericity result reads as follows.

THEOREM 3.13. *Let M, V and F be as above and let S be the image of the zero section in $T^*\mathcal{M}$. If $F \pitchfork S$ then for a generic $A \in V$ the critical points of the smooth function $f_A : \mathcal{M} \rightarrow \mathbb{R}$ are non-degenerate.*

Proof. Fix $A \in V$ and define

$$F_A : \mathcal{M} \rightarrow T^*\mathcal{M}, \quad F_A(P) := F(A, P) \quad (3.48)$$

From the Transversality Theorem 3.12 it follows that the set

$$R := \{A \in V \mid F_A \pitchfork S\} \quad (3.49)$$

is open and dense in V if $F \pitchfork S$. In the following, we will prove that $F_A \pitchfork S$ is equivalent to the fact that the Hessian of f_A is non-degenerate in the critical points. This will prove the theorem.

First, notice that $P_c \in F_A^{-1}(S)$ if and only if $P_c \in \mathcal{M}$ is a critical point of f_A . Therefore, the transversality condition

$$\text{Im } T_{P_c} F_A + T_{F_A(P_c)} S = T_{F_A(P_c)} T^* \mathcal{M}, \quad (3.50)$$

is equivalent to

$$\text{Im } T_{P_c} F_A + T_0 S = T_0 T^* \mathcal{M}. \quad (3.51)$$

To rewrite this condition (3.51) in local coordinates, let $\varphi : U \rightarrow W \subset T_{P_c} \mathcal{M}$ be a chart on an open subset $U \subset \mathcal{M}$ around P_c such that $\varphi^{-1}(0) = P_c$ and $D\varphi^{-1}(0) = \text{id}$. Then define

$$\tilde{f}_A := f_A \circ \varphi^{-1} : W \rightarrow \mathbb{R}. \quad (3.52)$$

Moreover, φ induces a chart $\psi : \pi^{-1}(U) \rightarrow W \times T_{P_c}^* \mathcal{M} \subset T_{P_c} \mathcal{M} \times T_{P_c}^* \mathcal{M}$ around $F_A(P_c)$ via

$$\psi(\gamma) = (x, (D\varphi^{-1}(x))^*(\gamma)), \quad x := \varphi \circ \pi(\gamma), \quad (3.53)$$

Here, $\pi : T^* \mathcal{M} \rightarrow \mathcal{M}$ refers to the natural projection and $(D\varphi^{-1}(x))^*(\gamma) := \gamma \circ D\varphi^{-1}(x)$. Thus, for

$$\tilde{F}_A := \psi \circ F_A \circ \varphi^{-1} : W \rightarrow W \times T_{P_c}^* \mathcal{M} \quad (3.54)$$

one has $\tilde{F}_A(x) = (x, d\tilde{f}_A(x))$. Since transversality of F_A to S is preserved in local coordinates, (3.51) is equivalent to

$$\text{Im } D\tilde{F}_A(0) + T_{P_c} \mathcal{M} \times \{0\} = T_{P_c} \mathcal{M} \times T_{P_c}^* \mathcal{M}. \quad (3.55)$$

Then $D\tilde{F}_A(0) = (\text{id}, d^2 \tilde{f}_A(0))$ yields that (3.55) is fulfilled if and only if $d^2 \tilde{f}_A(0)$ is nonsingular. Finally, the conclusion follows from the identity $\text{Hess}_{f_A}(P_c) = d^2 \tilde{f}_A(0)$ which is satisfied due to the fact that P_c is a critical point and $D\varphi^{-1}(0) = \text{id}$. Here, $\text{Hess}_{f_A}(P_c)$ denotes the Hessian form corresponding to the Hessian operator via $\text{Hess}_{f_A}(P_c)(x, y) = \langle \mathbf{H}_{f_A}(P_c)x, y \rangle$ for all $x, y \in T_{P_c} \mathcal{M}$. \square

For the generalized Rayleigh-quotient, we obtain the following result.

COROLLARY 3.14. *The critical points of the generalized Rayleigh-quotient are generically non-degenerate.*

Proof. Set $\mathcal{M} := \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, $V := \mathfrak{hct}_N$. For the simplicity, we will identify the cotangent bundle $T^* \mathcal{M}$ with the tangent bundle $T \mathcal{M}$ and work with the map

$$F : V \times \mathcal{M} \rightarrow T \mathcal{M}, \quad (A, P) \mapsto \text{grad} \rho_A(P), \quad (3.56)$$

instead of (3.47), where $\text{grad} \rho_A(P)$ is the Riemannian gradient of ρ_A at P . We will show that $F \pitchfork S$, where S is now the image of the zero section in $T \mathcal{M}$, i.e.

$$\text{Im } T_{(A,P)} F + T_{F(A,P)} S = T_{F(A,P)} T \mathcal{M}, \quad (3.57)$$

for all $(A, P) \in V \times \mathcal{M}$ with $\text{grad} \rho_A(P) = 0$. As in the proof of Theorem 3.13, we rewrite the transversality condition (3.57) in local coordinates, i.e.

$$\text{Im } D\tilde{F}(A, 0) + T_P \mathcal{M} \times \{0\} = T_P \mathcal{M} \times T_P \mathcal{M}, \quad (3.58)$$

where

$$\tilde{F} := \psi \circ F \circ (\text{id} \times \varphi^{-1}) : V \times W \rightarrow W \times \text{T}_P \mathcal{M}. \quad (3.59)$$

Here, $\varphi : U \rightarrow W \subset \text{T}_P \mathcal{M}$ is a chart around P with $\varphi^{-1}(0) = P$ and $D\varphi^{-1}(0) = \text{id}$ and $\psi : \pi^{-1}(U) \rightarrow W \times \text{T}_P \mathcal{M} \subset \text{T}_P \mathcal{M} \times \text{T}_P \mathcal{M}$ is the corresponding induced chart around $F(A, P)$. With this choice of charts, we obtain

$$\tilde{F}(A, x) = (x, \nabla \tilde{\rho}_A(x)), \quad (3.60)$$

where $\tilde{\rho}_A := \rho_A \circ \varphi^{-1} : W \rightarrow \mathbb{R}$. Since $A \mapsto \tilde{\rho}_A(0)$ is linear, one has

$$D\tilde{F}(A, 0)(X, \xi) = \left(\xi, \nabla \tilde{\rho}_X(0) + d^2 \tilde{f}_A(0) \xi \right). \quad (3.61)$$

Thus, condition (3.58) holds if and only if

$$\text{Im } \nabla \tilde{\rho}_{(\cdot)}(0) + \text{Im } d^2 \tilde{f}_A(0) = \text{T}_P \mathcal{M}. \quad (3.62)$$

Finally, we will show that $\text{Im } \nabla \tilde{\rho}_{(\cdot)}(0) = \text{T}_P \mathcal{M}$ which clearly guarantees (3.62). Let $\xi := (\xi_1, \dots, \xi_r) \in (\text{Im } \nabla \tilde{\rho}_{(\cdot)}(0))^\perp$, then we obtain

$$\begin{aligned} 0 &= \langle \nabla \tilde{\rho}_X(0), \xi \rangle = d\tilde{\rho}_X(0)\xi = d\rho_X(P)\xi \\ &= \text{tr} \left(X \left(\sum_{j=1}^r P_1 \otimes \dots \otimes \xi_j \otimes \dots \otimes P_r \right) \right), \end{aligned} \quad (3.63)$$

for all $X \in \mathfrak{her}_N$. Notice, that the equality $d\tilde{\rho}_X(0)\xi = d\rho_X(P)\xi$ follows from $D\varphi^{-1}(0) = \text{id}$. Therefore,

$$\sum_{j=1}^r P_1 \otimes \dots \otimes \xi_j \otimes \dots \otimes P_r = 0 \quad (3.64)$$

and this holds if and only if $\xi_1 = 0, \dots, \xi_r = 0$, since all summands in (3.64) are orthogonal to each other. Thus, we have proved that $\tilde{F} \pitchfork \text{T}_P \mathcal{M} \times \{0\}$ and hence $F \pitchfork S$. From the Theorem 3.13 it follows immediately that the critical points of the generalized Rayleigh-quotient are generically non-degenerate. \square

Unfortunately, for best multilinear rank tensor approximation and subspace clustering problems, we cannot conclude from Corollary 3.14 that the critical points of ρ_A are generically non-degenerate. In these cases, the resulting matrices A are restricted to a thin subset of \mathfrak{her}_N and thus the genericity statement with respect \mathfrak{her}_N in Corollary 3.14 does not carry over straight-forwardly.

4. Numerical Methods. Exploiting the geometrical structure of the constraint set $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, we develop two numerical methods, a Newton-like and a conjugated gradient algorithm, for optimizing the generalized Rayleigh-quotient ρ_A , with $A \in \mathfrak{her}_N$, $N := n_1 n_2 \dots n_r$.

4.1. Newton-like algorithm. The intrinsic Riemannian Newton algorithm is described by means of the Levi-Civita connection taking iteration steps along geodesics [9, 29]. Sometimes geodesics are difficult to determine, thus, here we are interested in a more general approach, which introduces the Newton iteration via local coordinates, see [1, 13, 27]. More precisely, we follow the ideas in [13] and use a pair of local coordinates on $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, i.e. normal coordinates and QR-coordinates.

Recall that, a *local parametrization*^{*} of $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ around a point $P := (P_1, \dots, P_r)$ is a smooth map

$$\mu_P : \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) \rightarrow \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$$

satisfying the additional conditions

$$\mu_P(0) = P \quad \text{and} \quad D\mu_P(0) = \text{id}_{\text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})}. \quad (4.1)$$

Riemannian normal coordinates are given by the Riemannian exponential map

$$\mu_P^{\text{exp}}(\xi) = (e^{-[\xi_1, P_1]} P_1 e^{[\xi_1, P_1]}, \dots, e^{-[\xi_r, P_r]} P_r e^{[\xi_r, P_r]}), \quad (4.2)$$

while *QR-type coordinates* are defined by the QR-approximation of the matrix exponential, i.e.

$$\mu_P^{\text{QR}}(\xi) = ([X_1]_Q^\dagger P_1 [X_1]_Q, \dots, [X_r]_Q^\dagger P_r [X_r]_Q). \quad (4.3)$$

Here $[X_j]_Q$ denotes the Q -factor from the unique QR decomposition of $X_j := I + [\xi_j, P_j]$.

Now, let $P^* := (P_1^*, \dots, P_r^*) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ be a critical point of ρ_A . Choose $P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ in a neighborhood of P^* and perform the following Newton-like iteration

$$P^{\text{new}} := \mu_P^{\text{QR}}(\xi), \quad (4.4)$$

where $\xi := (\xi_1, \dots, \xi_r) \in \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is a solution of the Newton equation

$$\mathbf{H}_{\rho_A}(P)\xi = -\text{grad } \rho_A(P). \quad (4.5)$$

Replacing the objects in (4.5) by their explicit form computed in the previous section, we get the following Newton equation:

$$-\text{ad}_{P_j} \text{ad}_{\hat{A}_j} \xi_j + \sum_{k=1, k \neq j}^r \text{ad}_{P_j}^2 \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, \xi_k, \dots, P_r) = -\text{ad}_{P_j}^2 \hat{A}_j, \quad (4.6)$$

for all $j = 1, \dots, r$. As mentioned before, let $\hat{A}_j := \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_r)$. Solving this system in the embedding space $\mathfrak{het}_{n_1} \times \dots \times \mathfrak{het}_{n_r}$ requires a higher number of parameters than necessary. However, exploiting the particular structure of the tangent vectors

$$\xi_j = \Theta_j \zeta_j \Theta_j^\dagger = \Theta_j \begin{bmatrix} 0 & Z_j \\ Z_j^\dagger & 0 \end{bmatrix} \Theta_j^\dagger, \quad (4.7)$$

^{*}Clearly, one can define a local parametrization more generally, i.e. without requiring the second part of (4.1).

allows us to solve (4.6) with the minimum number of parameters equal to the dimension of $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$. Thus, by multiplying (4.6) from the left with Θ_j and from the right with Θ_j^\dagger , we obtain an equation in the variables $Z_j \in \mathbb{C}^{m_j \times (n_j - m_j)}$, i.e.

$$\Psi_j' Z_j - Z_j \Psi_j'' - \sum_{k=1, k \neq j}^r \Phi_j(Z_k) = \Psi_j''', \quad (4.8)$$

where the terms Ψ_j' , Ψ_j'' , Ψ_j''' and $\Phi_j(Z_k)$ are computed in the following. Let

$$\Theta_j = \begin{bmatrix} U_j & V_j \end{bmatrix} \in \text{SU}_{n_j}, \quad (4.9)$$

where U_j and V_j are $n_j \times m_j$ and $n_j \times (n_j - m_j)$ matrices, respectively. Then,

$$\Psi_j' = U_j^\dagger \hat{A}_j U_j, \quad \Psi_j'' = V_j^\dagger \hat{A}_j V_j, \quad \Psi_j''' = U_j^\dagger \hat{A}_j V_j. \quad (4.10)$$

For expressing $\Phi_j(Z_k)$ with $j < k$, we introduce the multilinear operators $\Psi_{A,j,k} : \mathfrak{her}_{n_1} \times \cdots \times \mathfrak{her}_{n_r} \rightarrow \mathbb{C}^{n_j \times n_k \times n_j \times n_k}$ defined in a similar way as $\Psi_{A,j}$ by

$$\text{tr} \left(A(X_1 \otimes \cdots \otimes X_j S \otimes \cdots \otimes X_k T \otimes \cdots \otimes X_r) \right) = \text{tr} \left(\Psi_{A,j,k}(X_1, \dots, X_r)(S \otimes T) \right), \quad (4.11)$$

for all $S \in \mathbb{C}^{n_j \times n_j}$ and $T \in \mathbb{C}^{n_k \times n_k}$. For convenience, we will use the following shortcut

$$\hat{A}_{jk} := \Psi_{A,j,k}(P_1, \dots, I_{n_j}, \dots, I_{n_k}, \dots, P_r) \in \mathfrak{her}_{n_j \cdot n_k}. \quad (4.12)$$

Furthermore, we partition the matrix \hat{A}_{jk} into block form

$$\hat{A}_{jk} =: [\hat{\mathbf{a}}_{st}]_{s,t=1}^{n_j}, \quad (4.15)$$

where each $\hat{\mathbf{a}}_{st}$ is an $n_k \times n_k$ matrix. Then, the linear map $\Phi_j : \mathbb{C}^{m_k \times (n_k - m_k)} \rightarrow \mathbb{C}^{m_j \times (n_j - m_j)}$ is given by

$$Z_k \mapsto \Phi_j(Z_k) := U_j^\dagger \left[\text{tr}(U_k^\dagger \hat{\mathbf{a}}_{st} V_k Z_k^\dagger + Z_k V_k^\dagger \hat{\mathbf{a}}_{st}^\dagger U_k) \right]_{s,t=1}^{n_j} V_j. \quad (4.16)$$

Finally, the complete Newton-like algorithm for the optimization of ρ_A on $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is given by Algorithm 1.

Suggestions for implementation. (a) For an arbitrary matrix $A \in \mathfrak{her}_N$, the computation of \hat{A}_j and \hat{A}_{jk} is performed according to formula (3.29). This can be simplified in the case of the applications described in Section 3.3.

Case 1. If $A = vv^\dagger$, with $v = \text{vec}(\mathcal{A})$, $\mathcal{A} \in \mathbb{C}^{n_1 \times \cdots \times n_r}$, then

$$\hat{A}_j = B_{(j)} \cdot B_{(j)}^\dagger \in \mathfrak{her}_{n_j} \quad \text{and} \quad \hat{A}_{jk} = C_{(j,k)} \cdot C_{(j,k)}^\dagger \in \mathfrak{her}_{n_j n_k}, \quad (4.17)$$

where $B_{(j)}$ and $C_{(j,k)}$ are the j -th mode and respectively (j,k) -th mode matrices of the tensors $\mathcal{B} = \mathcal{A} \times_1 U_1^\dagger \times_2 \cdots \times_k I_{n_k} \times_{k+1} \cdots \times_r U_r^\dagger$ and $\mathcal{C} = \mathcal{A} \times_1 U_1^\dagger \times_2 \cdots \times_j I_{n_j} \times_{j+1} \cdots \times_k I_{n_k} \times_{k+1} \cdots \times_r U_r^\dagger$, respectively.

Case 2. If $A = \underbrace{\sum_{l=1}^L x_l x_l^\dagger \otimes \cdots \otimes x_l x_l^\dagger}_{r \text{ times}}$ with $x_l \in \mathbb{C}^n$ and $L \in \mathbb{N}$, then

$$\hat{A}_j = \sum_{l=1}^L \left(\prod_{\substack{i=1 \\ i \neq j}}^r \|P_i x_l\|^2 \right) x_l x_l^\dagger \quad \text{and} \quad \hat{A}_{jk} = \sum_{l=1}^L \left(\prod_{\substack{i=1 \\ i \neq j, i \neq k}}^r \|P_i x_l\|^2 \right) x_l x_l^\dagger \otimes x_l x_l^\dagger. \quad (4.18)$$

ALGORITHM 1.

N-like algorithm

Step 1. Starting point: Given $P = (P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ choose

$$\Theta_j = \begin{bmatrix} U_j & V_j \end{bmatrix} \in \text{SU}_{n_j}, \quad U_j^\dagger U_j = I_{m_j}, \quad V_j^\dagger V_j = I_{n_j - m_j},$$

such that $P_j = \Theta_j \Pi_j \Theta_j^\dagger$, for $j = 1, \dots, r$.

Step 2. Stopping criterion: $\|\text{grad}_{\rho_A}(P)\|/\rho_A(P) < \varepsilon$.

Step 3. Newton direction: Set

$$\hat{A}_j := \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_r)$$

and compute $\Psi'_j, \Psi''_j, \Psi'''_j$ as in (4.10), for $j = 1, \dots, r$.

Set

$$\hat{A}_{jk} := \Psi_{A,j,k}(P_1, \dots, I_{n_j}, \dots, I_{n_k}, \dots, P_r)$$

and compute $\Phi_j(Z_k)$ as in (4.15) and (4.16), for $j, k = 1, \dots, r$, with $j < k$. Furthermore, set $\Phi_k(Z_j) = \Phi_j(Z_k)^\dagger$ and solve the Newton equation

$$\Psi'_j Z_j - Z_j \Psi''_j - \sum_{k=1, k \neq j}^r \Phi_j(Z_k) = \Psi'''_j, \quad (4.13)$$

to obtain $Z_j \in \mathbb{C}^{m_j \times (n_j - m_j)}$, for $j = 1, \dots, r$.

Step 4. QR-updates:

$$\Theta_j^{\text{new}} = \Theta_j \begin{bmatrix} I_{m_j} & -Z_j \\ Z_j^\dagger & I_{n_j - m_j} \end{bmatrix}_Q \quad \text{and} \quad P_j^{\text{new}} = \Theta_j \Pi_j \Theta_j^{\text{new}\dagger} \quad (4.14)$$

for all $j = 1, \dots, r$. Here $[\]_Q$ refers to the Q part from the QR factorization.

Step 5. Set $P := P^{\text{new}}, \Theta := \Theta^{\text{new}}$ and go to Step 2.

(b) To solve the system (4.8), one can rewrite it as a linear equation on \mathbb{R}^d (d is the dimension of $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$) using matrix Kronecker products and vec-operations, then solve this by any linear equation solver.

(c) The computation of geodesics on matrix manifolds usually requires the matrix exponential map, which is in general an expensive procedure of order $O(n^3)$. Yet, for the particular case of the Grassmann manifold $\text{Gr}_{m,n}$, Gallivan et.al. [10] have developed an efficient method to compute the matrix exponential, reducing the complexity order to $O(nm^2)$ ($m < n$). Our approach, however, is based on a first order approximation of the matrix exponential $e^{[\zeta, \Pi]}$ followed by a QR-decomposition to preserve orthogonality/unitarity. Explicitly, it is given by

$$\begin{bmatrix} I_m & -Z \\ Z^\dagger & I_{n-m} \end{bmatrix}_Q = W \begin{bmatrix} D^{-1} & \Sigma D^{-1} & 0 \\ -\Sigma^\dagger D^{-1} & D^{-1} & 0 \\ 0 & 0 & I_{n-2m} \end{bmatrix} W^\dagger, \quad (4.19)$$

where $Z = X\Sigma Y^\dagger$ with $X \in \text{SU}_m$, $Y \in \mathbb{C}^{(n-m) \times m}$, $Y_j^\dagger Y_j = I_{m_j}$ and $\Sigma \in \mathbb{C}^{m \times m}$ diagonal. Furthermore,

$$W := \begin{bmatrix} X^\dagger & 0 & 0 \\ 0 & Y & Y' \end{bmatrix} \in \text{SU}_n, \quad D := \sqrt{I_m + \Sigma^\dagger \Sigma}, \quad (4.20)$$

where $[Y \ Y'] \in \text{SU}_{n-m}$ is an unitary completion of Y . The computational complexity of this QR-factorization is of order $O((n-m)m^2)$.

(d) The convergence of the algorithm is not guaranteed for arbitrary initial conditions $P \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ and even in the case of convergence the limiting point need not be a local maximizer of the function. To overcome this, one could for example test if the computed direction is ascending, else take the gradient as the new direction. Furthermore, one can make an iterative line-search in the ascending direction.

In the following theorem we prove that the sequence generated by Algorithm 1 converges quadratically to a critical point of the generalized Rayleigh-quotient ρ_A if the sequence starts in a sufficiently small neighborhood of the critical point.

THEOREM 4.1. *Let $A \in \mathfrak{her}_N$ and $P^* \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ be a non-degenerate critical point of the generalized Rayleigh-quotient ρ_A , then the sequence generated by the N-like algorithm converges locally quadratically to P^* .*

Proof. For the critical point $P^* \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, the Riemannian coordinates (4.2) and the QR- coordinates (4.3) satisfy the condition $D\mu_{P^*}^{\text{exp}}(0) = D\mu_{P^*}^{\text{QR}}(0) = \text{id}_{T_{P^*}\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})}$. Thus, according to Theorem 4.1. from [13] there exists a neighborhood $V \subset \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ such that the sequence of iterates generated by the N-like algorithm converges quadratically to P^* when the initial point P is in V . \square

4.2. Riemannian conjugated gradient algorithm. The quadratic convergence of the Newton-like algorithm has the drawback of high computational complexity. Solving the Newton equation (4.8) yields a cost per iteration of order $O(d^3)$, where d is the dimension of $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$. In what follows, we offer as an alternative to reduce the computational costs of the Newton-like algorithm by a conjugated gradient method. The linear conjugated gradient (LCG) method is used for solving large systems of linear equations with a symmetric positive definite matrix, which is achieved by iteratively minimizing a convex quadratic function $x^\dagger Ax$. The initial direction d_0 is chosen as the steepest descent and every forthcoming direction d_j is required to be conjugated to all the previous ones, i.e. $d_j^\dagger A d_k = 0$, for all $k = 0, \dots, j-1$. The exact maximum along a direction gives the next iterate. Hence, the optimal solution is found in at most n steps, where n is the dimension of the problem. Nonlinear conjugated gradient (NCG) methods use the same approach for general functions $f : \mathbb{R}^n \rightarrow \mathbb{R}$, not necessarily convex and quadratic. The update in this case reads as

$$x^{\text{new}} = x + \alpha d \quad \text{and} \quad d^{\text{new}} = -\nabla f(x^{\text{new}}) + \beta d,$$

where the step-size α is obtained by a line search in the direction d

$$\alpha = \arg \min_t f(x + td) \quad (4.21)$$

and β is given by one of the formulas: Fletcher-Reeves, Polak-Ribiere, Hestenes-Stiefel, or other. We refer to [29] for the generalization of the NCG method to a Riemannian manifold. For the computation of the step-size along the geodesic in direction ξ , an exact line search — as in the classical case — is an extremely expensive

procedure. Therefore, one commonly approximates (4.21) by an Armijo-rule, which ensures at least that the step length decreases the function sufficiently. We, however, have decided to compute the step-size by performing a one-dimensional Newton-step along the geodesic, since in the neighborhood of a critical point one Newton step can lead very close to the solution. Therefore, at $P = (P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ the step-size in direction $\xi = (\xi_1, \dots, \xi_r) \in \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is given by

$$\alpha = -\frac{(\rho_A \circ \gamma)'(0)}{(\rho_A \circ \gamma)''(0)} \quad (4.22)$$

where $\gamma : I \rightarrow \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is the unique geodesic through P in direction ξ .

Let $\Theta := (\Theta_1, \dots, \Theta_r)$ be such that $\Theta_k P_k \Theta_k^\dagger = \Pi_k$. Furthermore, let $P^{\text{new}} := (P_1^{\text{new}}, \dots, P_r^{\text{new}})$ denote the updated point in $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ via the QR-coordinates as in (4.14). For the computation of the new direction, a “transport” of the old direction $\xi = (\xi_1, \dots, \xi_r)$ from $\text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ to the tangent space $\text{T}_{P^{\text{new}}} \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is required. We use the following approximation for the parallel transport of ξ along the geodesic through P in direction ξ

$$\xi_j \mapsto \Theta_j^{\text{new}} \begin{bmatrix} 0 & Z_j \\ Z_j^\dagger & 0 \end{bmatrix} \Theta_j^{\text{new}\dagger}, \quad \text{where } \Theta_j^{\text{new}} = \Theta_j \begin{bmatrix} I_{m_j} & -Z_j \\ Z_j^\dagger & I_{n_j-m_j} \end{bmatrix}_Q, \quad (4.26)$$

for all $j = 1, \dots, r$.

The complete Riemannian conjugated gradient is presented as Algorithm 2.

It is recommended to reset the search direction to the steepest descent direction after d iterations, i.e. $Z_k^{\text{new}} := -g_k^{\text{new}}$, $k = 1, \dots, r$, where d refers to the dimension of the manifold. For the maximization of the generalized Rayleigh-quotient the initial direction is $Z_k = g_k$ and the update $Z_k^{\text{new}} = g_k^{\text{new}} + \beta Z_k$.

The convergence properties of the NCG methods are in general difficult to analyze. Yet, under moderate supplementary assumptions on the cost function one can guarantee that the NCG converges to a stationary point [24]. It is expected that the proposed Riemannian conjugated gradient method has properties similar to those of the NCG.

4.3. Numerical experiments. In this section we run several numerical experiments suitable for the applications mentioned in Section 3.2, i.e. best rank approximation for tensors and subspace clustering, to test the Newton-like (N-like) and Riemannian conjugated gradient (RCG) algorithms. The algorithms were implemented in MATLAB on a personal notebook with 1.8 GHz Intel Core 2 Duo processor.

4.3.1. Best multilinear rank- (m_1, \dots, m_r) tensor approximation.. To test the performance of our algorithms we have considered several examples of large size tensors of order 3 and 4 with entries chosen from the standard normal distribution and estimated their best low-rank approximation. We have started with a truncated HOSVD ([20]) and performed several HOOI iterates before we run our N-like and RCG algorithms. Depending on the size of the tensor, the number of HOOI iterations necessary to reach the region of attraction of a stationary point $P^* \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$, ranges from 10 to 100. As stopping criterion we have chosen that the relative norm of the gradient $\|\text{grad}_{\rho_A}(P)\|/\rho_A(P)$ is approximately 10^{-13} .

Computational complexity. The computational complexity of the N-like method is determined by the computation of the Hessian and the solution of the Newton equation (4.13). Thus, for the best rank- (m, m, m) approximation of a $n \times n \times n$

ALGORITHM 2.

RCG algorithm

Step 1. Starting point: Given $P = (P_1, \dots, P_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ choose

$$\Theta_j = \begin{bmatrix} U_j & V_j \end{bmatrix} \in \text{SU}_{n_j}, \quad U_j^\dagger U_j = I_{m_j}, \quad V_j^\dagger V_j = I_{n_j - m_j},$$

such that $P_j = \Theta_j \Pi_j \Theta_j^\dagger$, for $j = 1, \dots, r$.

Initial direction: Set

$$\hat{A}_j := \Psi_{A,j}(P_1, \dots, I_{n_j}, \dots, P_r),$$

compute $\Psi'_j, \Psi''_j, \Psi'''_j$ as in (4.10) and take the steepest descent direction

$$Z_j = -g_j := -\Psi'''_j,$$

for $j = 1, \dots, r$. Denote $Z := (Z_1, \dots, Z_r)$, $g := (g_1, \dots, g_r)$.

Step 2. Stopping criterion: $\|\text{grad}_{\rho_A}(P)\|/\rho_A(P) < \varepsilon$.

Step 3. QR-updates:

$$\Theta_j^{\text{new}} = \Theta_j \begin{bmatrix} \alpha I_{m_j} & \alpha Z_j \\ -\alpha Z_j^\dagger & \alpha I_{n_j - m_j} \end{bmatrix}_Q, \quad P_j = \Theta_j \Pi_j \Theta_j^{\text{new}\dagger}, \quad (4.23)$$

with the step-size given by $\alpha = -a/(b+c)$, where

$$\begin{aligned} a &:= \sum_{j=1}^r \text{tr} \left(\Psi_j''' Z_j^\dagger \right), \quad b := \sum_{j=1}^r \text{tr} \left(\Psi_j' Z_j Z_j^\dagger - Z_j \Psi_j'' Z_j^\dagger \right), \\ c &:= \sum_{j=1}^{r-1} \sum_{k=j+1}^r \rho_A(P_1, \dots, \xi_j, \dots, \xi_k, \dots, P_r), \end{aligned}$$

for $j = 1, \dots, r$. The tangent vectors ξ_j are given in (4.7).

Step 4. Set $P := P^{\text{new}}$ and $\Theta := \Theta^{\text{new}}$.

Step 5. New direction: Update $\Psi'_j, \Psi''_j, \Psi'''_j$ as in (4.10) and compute the new direction

$$Z_j^{\text{new}} = -g_j^{\text{new}} + \beta Z_j, \quad g_j^{\text{new}} := \Psi_j''', \quad (4.24)$$

for $j = 1, \dots, r$. Here, β is given by the Polak-Ribiere formula

$$\beta = \frac{\langle g^{\text{new}}, g^{\text{new}} - g \rangle}{\langle g, g \rangle} \quad (4.25)$$

Step 6. Set $g := g^{\text{new}}$, $Z := Z^{\text{new}}$ and go to Step 2.

tensor, the computation of the Hessian is dominated by tensor-matrix multiplications and is of order $O(n^3 m)$. Solving the Newton equation by Gaussian elimination gives a computational complexity of order $O(m^3(n-m)^3)$, i.e. the dimension of the manifold

to the power of three. For the computational costs of the RCG method we have to take into discussion only tensor-matrix multiplications, which give a cost per RCG iteration of order $O(n^3m)$.

Experimental results and previous work. The problem of best low-rank tensor approximation has enjoyed a lot of attention recently. Apart from the well known higher order orthogonal iterations – HOOI ([21]), various algorithms which exploit the manifold structure of the constraint set have been developed. We refer to [8, 18] for Newton methods, to [28] for quasi-Newton methods and to [17] for conjugated gradient and trust region methods on the Grassmann manifold. Similar to the Newton methods in [8, 18], our N-like method converges quadratically to a stationary point of the generalized Rayleigh-quotient when starting in its neighborhood.

We have compared our algorithms with the existing ones in the literature: quasi-Newton with BFGS, Riemannian conjugated gradient method which uses the Armijo-rule for the computation of the step-size (CG-Armijo), and HOOI. The algorithms were run on the same platform, identically initialized and with the same stopping criterion. For the BFGS quasi-Newton and limited memory quasi-Newton (L-BFGS) methods we have used the code available in [25].

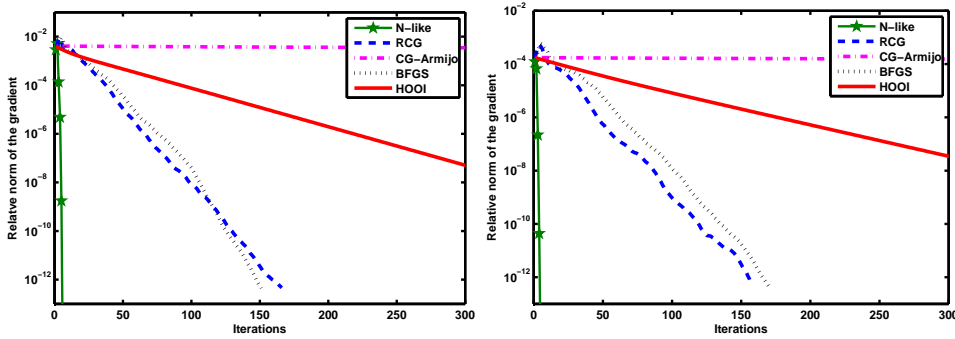


FIG. 4.1. Convergence for multilinear rank tensor approximation: number of iterations versus the relative norm of the gradient $\|\text{grad}_{\rho_A}(P^n)\|/\rho_A(P^n)$ at a logarithmic scale. Left: $100 \times 100 \times 100$ tensor approximated by a rank-(5, 5, 5) tensor. Right: $100 \times 150 \times 200$ tensor approximated by a rank-(15, 10, 5) tensor.

Fig. 4.1 shows convergence results for two large size tensors $100 \times 100 \times 100$ and $100 \times 150 \times 200$ approximated by rank-(5, 5, 5) and rank-(15, 10, 5) tensors, respectively. In Fig. 4.2 we plot the convergence behavior of the RCG method for the best rank-(10, 10, 10) approximation of a $200 \times 200 \times 200$ tensor (left) and for the best rank-(5, 5, 5, 5) approximation of a $50 \times 50 \times 50 \times 50$ tensor. Due to the limited memory space, we were not able to run the N-like and BFGS quasi-Newton algorithms for the example on the left. Yet it was still possible to run RCG, L-BFGS, CG-Armijo and HOOI.

In Table 4.1 we display the average CPU times necessary to compute a low rank best approximation for tensors of different sizes and orders by N-like, RCG, BFGS and L-BFGS quasi-Newton methods. We have performed 100 runs for each example.

Resume. First we mention that there is no guarantee that the N-like and RCG iterations converge to a local maximizer of the generalized Rayleigh-quotient. However, in the examples presented in Fig.4.1 and Fig.4.2 the limiting points are local maximizers. As the numerical experiments have shown, the N-like method has the advantage of fast convergence. Unfortunately, for large scale problems, the N-

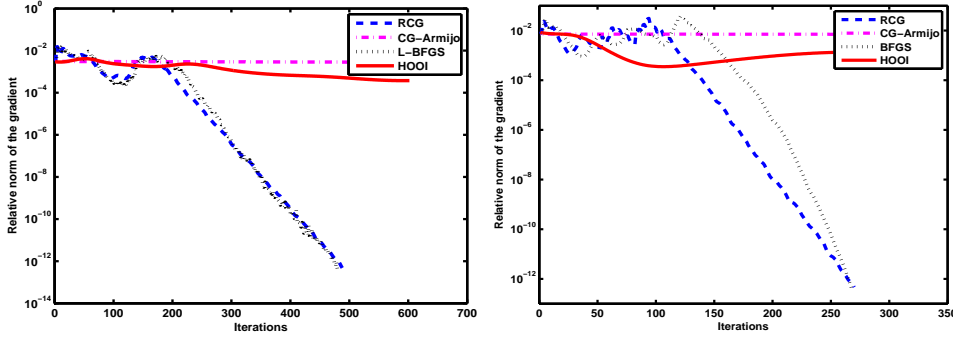


FIG. 4.2. Convergence for multilinear rank tensor approximation: number of iterations versus the relative norm of the gradient $\|\text{grad}_{\rho_A}(P^n)\|/\rho_A(P^n)$ at a logarithmic scale. Left: $200 \times 200 \times 200$ tensor approximated by a rank-(10, 10, 10) tensor. Right: $50 \times 50 \times 50 \times 50$ tensor approximated by a rank-(5, 5, 5, 5) tensor.

TABLE 4.1
Average CPU Time

Tensor size and rank	N-like	RCG	BFGS	L-BFGS
$50 \times 50 \times 50$, rank-(7, 8, 5)	2 s	6 s	24 s	13 s
$100 \times 100 \times 100$, rank-(5, 5, 5)	70 s	75 s	150 s	94 s
$200 \times 200 \times 200$, rank-(5, 5, 5)	-	11 min	-	14 min
$50 \times 50 \times 50 \times 50$, rank-(5, 5, 5, 5)	-	9 min	11 min	-

like algorithm can not be applied, as mentioned before. Even when it is possible to apply N-like algorithm, it needs a large amount of time per iteration. As an example, for the best rank-(10, 10, 10) of a $180 \times 180 \times 180$ tensor, one N-like iteration took three minutes. Related algorithms which explicitly compute the Hessian and solve the Newton equation, such as [8, 18], and those which approximately solve the Newton equation such as the trust region method [17], face the same difficulty for large scale problems. On the other hand, the low cost iterations of the RCG method makes it a good candidate to solve large size problems. The convergence rate is comparative to that of the BFGS quasi-Newton method in [25], but at much lower computational costs. Our experiments exhibit the shortest CPU time for the RCG method. In the examples in which the tensor was a small perturbation of a low-rank tensor, the RCG algorithm exhibits quadratic convergence.

4.3.2. Subspace Clustering. The experimental setup consists in choosing r subspaces in \mathbb{R}^3 ($r = 2, 3$ and 4) and collections of 200 randomly chosen[†] points on each subspace. Then, the sample points are perturbed by adding zero-mean Gaussian noise with standard deviation varying from 0% to 5% in the different experiments. Now, the goal is to detect the exact subspaces or to approximate them as good as possible. For this purpose, we apply our N-like and RCG algorithms to solve the associated optimization task, cf. Section 3.1. The error between the exact subspaces

[†]The points have been generated by fixing an orthogonal basis within the subspaces and choosing corresponding coordinates randomly with a uniform distribution over the interval $[-5, 5]$.

and the estimated ones is measured as in [33], i.e.

$$\text{err} := \frac{1}{r} \sum_{j=1}^r \arccos \left(\frac{1}{m_j^2} |\text{tr}(P_j \tilde{P}_j)| \right), \quad (4.27)$$

where P_j is the orthogonal projector corresponding to the exact subspace and \tilde{P}_j the orthogonal projector corresponding to the estimated one.

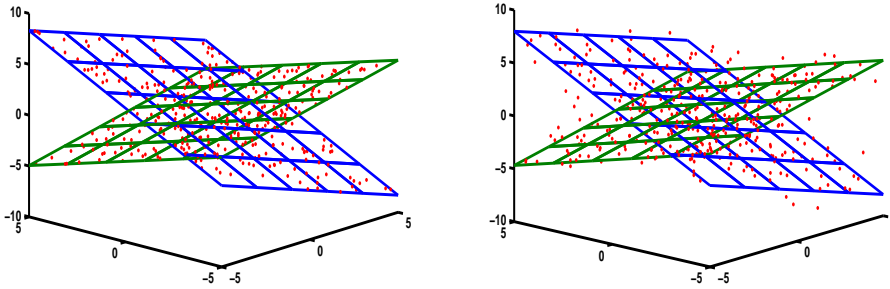


FIG. 4.3. Left: Data points drawn from the union of two subspaces of dimension 2 (through the origin) of \mathbb{R}^3 . Right: Data points from the left figure slightly perturbed by zero mean Gaussian noise with 5% standard deviation.

It can be easily checked that in the case of unperturbed data there is a unique non-degenerate minimizer of ρ_A , and it yields the exact subspaces. Thus, we expect that for noisy data the global minimizer still gives a good approximation. Since ρ_A has many local optima, for an arbitrary starting point our algorithms can converge to stationary points which lead to a significant error between the exact subspaces and their approximation. Thus, in what follows, we briefly describe a method (PDA, see below) for computing a suitable initial point which guarantees the convergence of our algorithms towards a good approximation of the exact subspaces in our numerical experiment:

The Polynomial Differential Algorithm (PDA) was proposed in [33]. It is a purely algebraic method for recovering a finite number of subspaces from a set of data points belonging to the union of these subspaces. From the data set finitely many homogeneous polynomials are computed such that their zero set coincides with the union of the sought subspaces. Then, an evaluation of their derivatives at given data points yields successively a basis of the orthogonal complement of subspaces one is interested in. For noisy data, a slightly modified version of PDA [33] yields an approximation of the unperturbed subspaces. This “first” approximation turned out to be a good starting point for our iterative algorithms which significantly improved the approximation quality.

For each noise level we perform 500 runs of the N-like and Local-CG algorithms for different data sets and compute the mean error between the exact subspaces and the computed approximations. As a preliminary step, we normalize all data points, such that no direction is favored.

In Fig. 4.3, 400 randomly chosen data points which lie exactly in the union of two

2-dimensional subspaces of \mathbb{R}^3 (left) and their perturbed[‡] images (right) are depicted. Moreover, the two plots display the exact subspaces (left) as well as the ones computed by our N-like algorithm (right). The error between the exact subspaces and our approximation is ca. 2° , whereas the error for the PDA approximation is ca. 5° .

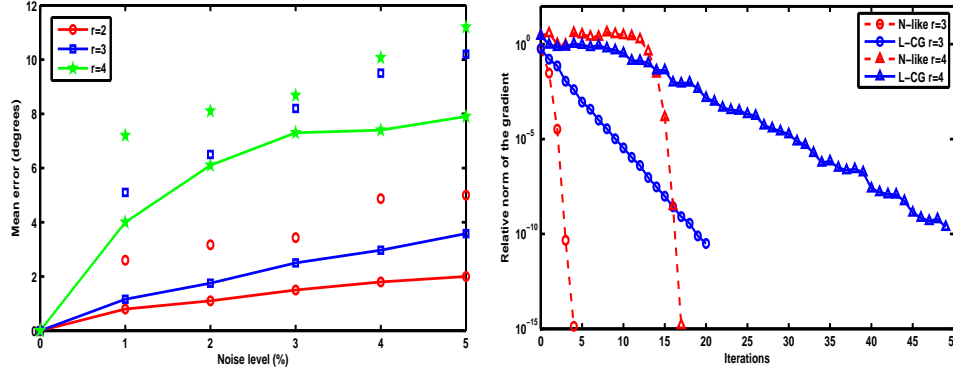


FIG. 4.4. *Left: The mean error for noise levels from 0% to 5% and different number of subspaces. The disconnected symbols refer to the initial error (PDA) and the corresponding continuous lines refer to the error estimated by our algorithms. Right: Convergence of N-like and RCG for subspace clustering: number of iterations versus the relative norm of the gradient $\|\text{grad}_{\rho_A}(P^n)\|/\rho_A(P^n)$ at a logarithmic scale. Data points from 3 and resp. 4 subspaces perturbed with 5% Gaussian noise. Average CPU time: ca. 0.4 and ca. 2 seconds for the N-like and RCG algorithm, respectively (1.8 GHz Intel Core 2 Duo processor).*

In Fig. 4.4, we plot the mean error (left) for different noise levels and different number of subspaces. We have included also the mean error for the starting point of our algorithms, i.e. for the PDA approximation. On the right we demonstrate the fast convergence rate of the N-like and RCG algorithms for the case of 3 and, respectively, 4 subspaces.

Resume. Our numerical experiments have proven that (i) the minimization task proposed in Section 3 is capable to solve subspace detection problems and (ii) our numerical algorithms initialized with the PDA starting point yield an effective method for computing a reliable approximation of the perturbed subspaces. How the approximation of the perturbed subspaces varies when the noise in the data follows some law of distribution, is the subject of future investigation.

5. Appendix.

Here we provide a proof of Proposition 2.3, which states that there exists a global Riemannian isometry φ between $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$ and $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$.

Proof. The surjectivity of φ is clear from the definition of $\text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n})$. To prove the injectivity of φ we use induction over r . Choose $(P_1, \dots, P_r), (Q_1, \dots, Q_r) \in \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ such that $P_1 \otimes \dots \otimes P_r = Q_1 \otimes \dots \otimes Q_r$, i.e.

$$\alpha_{ij} P_r = \beta_{ij} Q_r \quad \text{for all } i, j, \quad (5.1)$$

where α_{ij} and β_{ij} are the entries of $P_1 \otimes \dots \otimes P_{r-1}$ and $Q_1 \otimes \dots \otimes Q_{r-1}$, respectively. Thus it exists $\gamma \in \mathbb{C}$ such that $P_r = \gamma Q_r$. Since P_r and Q_r have only 0 and 1 as eigenvalues it follows that $\gamma = 1$ and $P_r = Q_r$. Therefore, $P_1 \otimes \dots \otimes P_r = Q_1 \otimes \dots \otimes Q_r$

[‡]Gaussian noise with 5% standard deviation

implies that

$$P_1 \otimes \cdots \otimes P_{r-1} = Q_1 \otimes \cdots \otimes Q_{r-1} \quad (5.2)$$

and the procedure can be repeated until we obtain $P_j = Q_j$, for all $j = 1, \dots, r$. Thus the injectivity of φ is proven. So φ is a continuous bijective map with continuous inverse φ^{-1} due to the compactness of $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$. Moreover, the map φ is smooth since the components of $P_1 \otimes \cdots \otimes P_r$ are polynomial functions. Let $P := (P_1, \dots, P_r)$ and $\mathbf{P} := P_1 \otimes \cdots \otimes P_r$. Consider the tangent map of φ at P , i.e.

$$\begin{aligned} D\varphi(P) : \text{T}_P \text{Gr}^{\times r}(\mathbf{m}, \mathbf{n}) &\rightarrow \text{T}_P \text{Gr}^{\otimes r}(\mathbf{m}, \mathbf{n}), \\ (\xi_1, \dots, \xi_r) &\mapsto \sum_{j=1}^r P_1 \otimes \cdots \otimes \xi_j \otimes \cdots \otimes P_r. \end{aligned} \quad (5.3)$$

With the inner products (2.11) and (2.12) defined on \mathfrak{her}_N and $\mathfrak{her}_{n_1} \times \cdots \times \mathfrak{her}_{n_r}$, respectively, one has

$$\langle D\varphi(P)\xi, D\varphi(P)\eta \rangle = \sum_{j=1}^r M_j \text{tr}(\xi_j \eta_j) = \langle \xi, \eta \rangle, \quad M_j := \prod_{k=1, k \neq j}^r m_k. \quad (5.4)$$

This implies that the tangent map $D\varphi(P)$ is a linear isometry. Thus, it is invertible and therefore φ is a local diffeomorphism. Moreover, since φ is bijective it is a global diffeomorphism, giving thus a global Riemannian isometry when the metric on $\text{Gr}^{\times r}(\mathbf{m}, \mathbf{n})$ is defined by (2.14). \square

Acknowledgements. This work has been supported by the Federal Ministry of Education and Research (BMBF) through the project FHprofUnd 2007: Cooperation Program between Universities of Applied Science and Industry, "Development and Implementation of Novel Mathematical Algorithms for Identification and Control of Technical Systems".

REFERENCES

- [1] P. A. ABSIL, R. MAHONY, AND R. SEPULCHRE, *Optimization algorithms on matrix manifolds*, Princeton University Press, 2007.
- [2] R. BHATIA, Partial traces and entropy inequalities, *Linear Algebra and its Applications*, 370, (2003), pp. 125–132.
- [3] R. W. BROCKETT, Least squares matching problems, *Linear Algebra and its Applications*, 122–124, (1989), pp. 761–777.
- [4] P. COMON, Tensor decompositions: State of the art and applications, *Mathematics in Signal Processing*, V, Oxford University Press, Oxford, (2002), pp. 1–24.
- [5] O. CURTEF, G. DIRR AND U. HELMKE, Riemannian optimization on tensor products of Grassmann manifolds: Applications to generalized Rayleigh-quotients manifolds, arXiv:1005.4854, (2011).
- [6] G. DIRR, U. HELMKE, M. KLEINSTEUBER AND TH. SCHULTE-HERBRÜGGEN, Relative C-numerical ranges for applications in quantum control and quantum information, *Linear and Multilinear Algebra*, 56, (2008), pp. 27–51.
- [7] C. ECKART, AND G. YOUNG, The approximation of one matrix by another of lower rank, *Psychometrika*, 1, (1936), pp. 211–263.
- [8] L. ELDEN, AND B. SAVAS, A Newton-Grassmann method for computing the best multilinear rank- (r_1, r_2, r_3) approximation of a tensor, *SIAM J. Matrix. Anal. Appl.*, 31(2), (2009), pp. 248–271.
- [9] D. GABAY, Minimizing a differentiable function over a differentiable manifold, *Journal of Optimization Theory and Applications*, 37(2), (1982), pp. 177–21.

- [10] K. A. GALLIVAN, A. SRIVASTAVA, X. LIU, AND P. VAN DOOREN, Efficient algorithms for inferences on Grassmann manifolds, *Proceedings of IEEE Conference on Statistical Signal Processing*, (2003), pp. 315–318.
- [11] G. H. GOLUB, *Matrix computations*, Johns Hopkins University Press, Baltimore, Maryland, 3rd edition, 1996.
- [12] W. GREUB, *Multilinear algebra*, Springer-Verlag, New York, 1978.
- [13] U. HELMKE, K. HÜPER AND J. TRUMF, Newton’s method on Grassmann manifolds, arXiv: 0709.2205v2, (2007).
- [14] U. HELMKE AND J. B. MOORE, *Optimization and dynamical systems*, Springer-Verlag London, 1994.
- [15] M. W. HIRSCH, *Differential topology*, Springer-Verlag, New York, 1976.
- [16] J. HO, M. H. YANG, J. LIM, K. C. LEE AND D. KRIEGMAN, Clustering appearances of objects under varying illumination conditions, *CVPR*, (2003), pp. 11–18.
- [17] M. ISHTEVA, L. DE LATHAUWER, P. A. ABSIL, AND S. VAN HUFFEL, On the best low multilinear rank approximation of higher-order tensors, based on trust region scheme, Technical report: ESAT-SISTA-09-142, (2009).
- [18] M. ISHTEVA, L. DE LATHAUWER, P. A. ABSIL, AND S. VAN HUFFEL, Differential-geometric Newton method for the best rank- (R_1, R_2, R_3) approximation of tensors, *Numerical Algorithms*, 51(2), (2009), pp. 179–194. Tributes to Gene H. Golub Part II.
- [19] S. LANG, *Algebra*, Rev. 3rd ed., Graduate Texts in Mathematics, 211, Springer-Verlag, New York, 2002.
- [20] L. DE LATHAUWER, B. DE MOOR, AND J. VANDEWALLE, A multilinear singular value decomposition, *SIAM J. Matrix Anal. Appl.*, 21(4), (2000), pp. 1253–1278.
- [21] L. DE LATHAUWER, B. DE MOOR, AND J. VANDEWALLE, On the best rank-1 and rank- (R_1, R_2, \dots, R_N) approximation of higher-order tensors, *SIAM J. Matrix Anal. Appl.*, 21(4), 2000, pp. 1324–1342.
- [22] R. MAHONY, U. HELMKE, AND J. B. MOORE, Gradient algorithms for principal component analysis, *J. Austral. Math. Soc. Ser.*, B37, (1996), pp. 430–450.
- [23] NIELSEN AND CHUANG, *Quantum computation and quantum information*, Cambridge University Press, 2000.
- [24] J. NOCEDAL AND S. J. WRIGHT, *Numerical optimization*, Springer-Verlag New York, 2nd edition, 2006.
- [25] B. SAVAS, Algorithm package manual: Best low rank tensor approximation, Department of Mathematics, Linköping University, Linköping, Sweden, (2008). (<http://www.mai.liu.se/~besav/soft.html>)
- [26] B. SAVAS, AND L. -H. LIM, Quasi-Newton methods on Grassmannians and multilinear approximations of tensors, *SIAM J. Sci. Comput.*, 32(6), (2010), pp. 3352–3393.
- [27] M. SHUB, Some remarks on dynamical systems and numerical analysis, *Dynamical systems and partial differential equations (Caracas)*, (1986), pp. 69–91.
- [28] V. DE SILVA, L. -H. LIM, Tensor rank and the ill-posedness of the best low-rank approximation problem, *SIAM J. Matrix. Anal. Appl.*, 30(3), (2008), pp. 1084–1127.
- [29] T. SMITH, Optimization techniques on Riemannian manifolds, *Fields Institute Communications*, 3, (1994).
- [30] M. TIPPING AND C. BISHOP, Mixtures of probabilistic principal component analyzers, *Neural Computation*, 11(2), (1999).
- [31] L. R. TUCKER, Some mathematical notes of three-mode factor analysis, *Psychometrika*, 31, (1966), pp. 279–311.
- [32] C. UDRISTE, *Convex functions and optimization methods on Riemannian manifolds*, Kluwer Academic Publishers, Dordrecht, 1994.
- [33] R. VIDAL, Y. MA, AND J. PIAZZI, A new GPCA algorithm for clustering subspaces fitting, differentiating and dividing polynomials, *CVPR*, (2004).
- [34] T. WEI, AND P. GOLDBART, Geometric measure of entanglement and applications to bipartite and multipartite quantum states, *Phys. Rev. A*, 68(042307), (2003).