AN INTRINSIC CG ALGORITHM FOR COMPUTING DOMINANT SUBSPACES

M. Kleinsteuber and K. Hüper

National ICT Australia, Canberra Research Laboratory Locked Bag 8001, Canberra ACT 2601, Australia and Department of Information Engineering, RSISE The Australian National University Canberra ACT 0200, Australia martin.kleinsteuber@nicta.com.au, knut.hueper@nicta.com.au

ABSTRACT

In this paper, a conjugate gradient method on the complex Graßmann manifold is proposed that computes the k-principal components of a Hermitian $(n \times n)$ -matrix. The algorithm is at most of order $\mathcal{O}(n^2k)$ and yields locally good convergence results.

Index Terms— Conjugate gradient methods, Principal component analysis, Block diagonalization, Hermitian matrices, Graßmann manifold

1. INTRODUCTION

A typical subproblem in array signal processing is the task of estimating the direction of arrival of signals. In order to determine the direction of arrival of k signals, as a subtask one has to determine the dominant subspace of the complex correlation matrix arising from data samples. This paper proposes an algorithm that computes this so-called signal space. In contrast to established algorithms (cf. e.g. [1, 8, 9]), the method here is based on the concept of geometric optimization. Exploiting the intrinsic structure of the problem lets us expect that the resulting algorithm is very accurate and robust to rounding errors. Following the ideas of Gabay (1982) [2], see also Smith (1994) [6], a conjugate gradient (CG) method on the complex Graßmann manifold is presented that computes the subspace corresponding to the k largest eigenvalues of a Hermitian matrix. A one-to-one implementation of the algorithms in [2] and [6], however, leads to far too expensive algorithms. Computing the geodesics or the parallel transport of a tangent vector on the complex Graßmann manifold requires for each algorithmic step the eigenvalue decomposition of a skew-Hermitian matrix having the same size as the correlation matrix itself and hence might be of higher complexity than the actual problem. Moreover, an exact computation of the step-size is in practice often out of reach. Here, we modify and specify the algorithm by using a second order approximation of the Riemannian exponential via OR-decompositions and a step size which is cheap to compute. Ultimately, this leads to an algorithm that maintains the geometric structure of the problem and is cheap.

2. PRELIMINARIES

Denote the set of unitary matrices by

$$\mathcal{U}_n := \{ U \in \mathbb{C}^{n \times n} \mid U^{\dagger} U = I_n \}, \tag{1}$$

where $(\cdot)^{\dagger}$ denotes conjugate transpose and I_n is the $(n \times n)$ -identity matrix. Furthermore, let

$$\mathfrak{u}_n := \{ \Omega \in \mathbb{C}^{n \times n} \mid \Omega^{\dagger} = -\Omega \}$$

$$\tag{2}$$

be the set of skew-Hermitian matrices. Our CG algorithm is based on the maximization task of a cost function on the complex Graßmann manifold, which we identify with the set of all rank k Hermitian projectors, i.e.

$$Gr_{k,n} := \{ UNU^{\dagger} \mid U \in \mathcal{U}_n \}, \tag{3}$$

where $N = \begin{bmatrix} I_k & 0 \\ 0 & 0 \end{bmatrix}$. We refer to [4] for the construction of a natural diffeomorphism between $Gr_{k,n}$ defined as above and the set of all k-dimensional complex subspaces of \mathbb{C}^n and the fact that $Gr_{k,n}$ can be considered as a submanifold of the set of $(n \times n)$ Hermitian matrices. The tangent space at $P \in Gr_{k,n}$ is given by

$$T_P Gr_{k,n} = \{ [P, \Omega] \mid \Omega \in \mathfrak{u}_n \}$$
(4)

with matrix commutator [A, B] := AB - BA. Endowing $T_P Gr_{k,n}$ with the inner product

$$g: T_P Gr_{k,n} \times T_P Gr_{k,n} \to \mathbb{R}, \quad (\xi, \eta) \mapsto \Re \operatorname{tr}(\xi\eta) \tag{5}$$

turns $Gr_{k,n}$ into a Riemannian manifold. The geodesic through P in direction ξ and hence the Riemannian exponential map are given by

$$\gamma_{\xi}(t) = e^{t[\xi, P]} P e^{-t[\xi, P]}, \quad \text{where } \xi \in T_P Gr_{k, n}.$$
(6)

The parallel transport of $\eta \in T_P Gr_{k,n}$ with respect to the Levi-Civita connection along the geodesic $\gamma_{\xi}(t)$ is given by

$$\eta(t) = e^{t[\xi, P]} \eta e^{-t[\xi, P]}.$$
(7)

3. SECOND ORDER APPROXIMATION OF THE RIEMANNIAN EXPONENTIAL

CG-methods for maximizing a smooth function in \mathbb{R}^n or \mathbb{C}^n are well established tools in numerical optimization, cf. [5]. Their generalizations to Riemannian manifolds, however, require the concept of

National ICT Australia is funded by the Australian Government's Department of Communications, Information Technology and the Arts and the Australian Research Council through *Backing Australia's Ability* and the ICT Research Centre of Excellence programs.

the Riemannian exponential function, [2, 6]. In this section, we propose a second order approximation for the Graßmannian case that, in contrast to a Taylor Series approach for approximating matrix exponentials, maintains the structure of the problem, i.e. ensures in a simple way that each iterate stays on the manifold.

Let \mathcal{R}_n be the set of complex invertible upper triangular $(n \times n)$ matrices with real but positive entries on the diagonal and let $\mathcal{G}l_n := \{X \in \mathbb{C}^{n \times n} \mid X \text{ is invertible}\}$. It follows from the Gram-Schmidt orthogonalization procedure that the map

$$\mathcal{U}_n \times \mathcal{R}_n \to \mathcal{G}l_n, \quad (Q, R) \mapsto QR$$
 (8)

is a diffeomorphism onto. According to Eq. (8) every $X \in \mathcal{G}l_n$ decomposes uniquely into

$$X =: X_Q X_R \tag{9}$$

with $X_Q \in \mathcal{U}_n$ and $X_R \in \mathcal{R}_n$. Furthermore, as an infinitesimal version of the above theorem, every complex $(n \times n)$ -matrix decomposes uniquely into the sum of a skew-Hermitian and an upper triangular matrix with real entries on the diagonal. We write

$$X =: X_{\text{skew}} + X_{\text{upp}} \tag{10}$$

with $X_{\text{skew}} \in \mathfrak{u}_n$ and X_{upp} upper triangular with real diagonal entries. The following lemma yields the desired approximation of the Riemannian exponential mapping. Note that our approach does *not* approximate the matrix exponential up to second order, only up to first order, but the Riemannian exponential mapping, defined by (6) is indeed approximated to second order.

Lemma 1. The map

$$q_{\Omega} \colon \mathbb{R} \to \mathcal{U}_n, \quad t \mapsto (I + t\Omega)_Q$$
 (11)

is smooth for all $\Omega \in \mathfrak{u}_n$. Moreover,

$$\dot{q}_{\Omega}(0) = \Omega, \quad \ddot{q}_{\Omega}(0) = (\Omega^2)_{\text{upp}}.$$
 (12)

Proof. For simplicity reasons we will drop the subscript Ω throughout the proof. As Ω is skew-Hermitian, all eigenvalues of $I + t\Omega$ have real part equal to 1 and hence $I + t\Omega$ is invertible. Therefore, q(t) as well as

$$r: \mathbb{R} \to \mathcal{R}, \quad t \mapsto (I + t\Omega)_R$$
 (13)

are smooth by the above observation. Now $I + t\Omega = q(t)r(t)$. Differentiating this equation with respect to t yields

$$\Omega = \dot{q}(t)r(t) + q(t)\dot{r}(t). \tag{14}$$

Since

$$r(0) = q(0) = I \tag{15}$$

it follows that

$$\Omega = \dot{q}(0) + \dot{r}(0).$$
(16)

Now $\dot{q}(0) \in \mathfrak{u}_n$ and $\dot{r}(0)$ is upper triangular and therefore

$$\dot{q}(0) = \Omega$$
 and $\dot{r}(0) = 0.$ (17)

For the computation of the second derivative we multiply Eq. (14) from the left with $q^{\dagger}(t)$ and from the right by $r^{-1}(t)$ to obtain

$$q^{\dagger}(t)\Omega r^{-1}(t) = q^{\dagger}(t)\dot{q}(t) + \dot{r}(t)r^{-1}(t).$$
(18)

Here the right hand side is again the unique decomposition into the sum of a skew-Hermitian and an upper triangular matrix with positive diagonal entries. It follows that

$$\dot{q}(t) = q(t) \left(q^{\dagger}(t) \Omega r^{-1}(t) \right)_{\text{skew}}$$
(19)

and hence using (15) and (17)

$$\begin{split} \ddot{q}(0) &= \frac{\mathrm{d}}{\mathrm{d}t} q(t) \left(q^{\dagger}(t) \Omega r^{-1}(t) \right)_{\mathrm{skew}} \Big|_{t=0} \\ &= \dot{q}(0) \left(q^{\dagger}(0) \Omega r^{-1}(0) \right)_{\mathrm{skew}} \\ &+ q(0) \left(\dot{q}^{\dagger}(0) \Omega r^{-1}(0) - q^{\dagger}(0) \Omega r^{-1}(0) \dot{r}(0) r^{-1}(0) \right)_{\mathrm{skew}} \\ &= \Omega \left(\Omega \right)_{\mathrm{skew}} + (-\Omega^2)_{\mathrm{skew}} = \Omega^2 - (\Omega^2)_{\mathrm{skew}} = (\Omega^2)_{\mathrm{upp}}. \end{split}$$

By the above lemma a second order accurate approximation of the geodesic (6) in a neighborhood of $P \in Gr_{k,n}$ now follows.

Corollary 2. Let $\xi \in T_P Gr_{k,n}$ and $q_{[\xi,P]}(t)$ defined as in Eq. (11). *The curve*

$$\alpha_{\xi} \colon \mathbb{R} \to Gr_{k,n}, \quad t \to q_{[\xi,P]}(t) P q_{[\xi,P]}(t)^{\dagger}$$
(20)

is a second order approximation of the geodesic (6) around P, i.e. the identities

$$\begin{aligned}
\alpha_{\xi}(0) &= \gamma_{\xi}(0), \\
\dot{\alpha}_{\xi}(0) &= \dot{\gamma}_{\xi}(0), \\
\ddot{\alpha}_{\xi}(0) &= \ddot{\gamma}_{\xi}(0)
\end{aligned}$$
(21)

hold.

4. THE CG-ALGORITHM

The CG method consists essentially of two steps. That is, line search in a given direction and the computation of the subsequent direction. For the task of maximizing a smooth function f on $Gr_{k,n}$ schematically we have the following.

CG-Sweep. Let $P_0 \in Gr_{k,n}$ be given. Set $H_0 := \nabla f(p_0)$ (the Riemannian gradient). Then for i = 0, ..., N - 1 ($N := \dim Gr_{k,n}$)

- (Line-Search) Compute $\lambda_i \in \mathbb{R}$ and set $P_{i+1} = \alpha_{H_0}(\lambda_i)$
- (Direction)
 Compute H_{i+1} according to a Riemannian adaption of the Hestenes-Stiefel-Formula or another formula known from the Euclidian case, cf. [5].

CG-Algorithm. Iterate the CG-sweeps.

4.1. Line Search

The CG-algorithm proposed in [2] and [6] requires an exact line search along $\gamma_{H_i}(t)$, i.e. finding the maximum of $f \circ \gamma_{H_i}(t)$ that is closest to 0. In practice, however, this is often not feasible, since either no closed form solution exists or, if it does, its computation is too expensive. Here, we replace the geodesic $\gamma_{H_i}(t)$ by the approximation (20), which is only given implicitly. Hence there will be no closed formula for an exact line search, no matter which function is to be optimized.

Nonetheless, since we consider our optimization task to be a local one, it is reasonable to assume that in an appropriate neighborhood of 0 the only critical point of $f \circ \alpha_{H_i}(t)$ is a local maximum. Therefore, we can expect that a one-dimensional Newton-step locally yields acceptable convergence results, i.e.

$$\lambda_i := -\frac{\frac{\mathrm{d}}{\mathrm{d}t}\Big|_{t=0} f \circ \alpha_{H_i}(t)}{\frac{\mathrm{d}^2}{\mathrm{d}t^2}\Big|_{t=0} f \circ \alpha_{H_i}(t)}.$$
(22)

Note, that for a strictly convex quadratic function in the Euclidian space, this corresponds to an exact line search.

4.2. Conjugate Directions

According to Eq. (7), the parallel transport along the approximation of the geodesics α_H , $H \in T_P Gr_{k,n}$ is given by

$$\tau\xi := q_{[H,P]}(\lambda_i)\xi q_{[H,P]}(\lambda_i)^{\dagger}$$
(23)

We shortly denote $\nabla f(P_i) = G_i$. The direction update is given by

$$H_{i+1} = -G_{i+1} + \gamma_i H_i,$$

where several choices for γ_i are common. In the following, we give manifold adaptions for the four most common formulas and propose one formula that seems to be new.

$$\gamma_i^{HS} = \frac{\Re \operatorname{tr}(G_{i+1}(G_{i+1} - \tau G_i))}{\Re \operatorname{tr}(\tau H_i(G_{i+1} - \tau G_i))} \qquad \text{(Hestenes-Stiefel)}$$
$$\gamma_i^{PR} = \frac{\Re \operatorname{tr}(G_{i+1}(G_{i+1} - \tau G_i))}{\|G_i\|^2} \qquad \text{(Polak-Ribière)}$$

$$\gamma_i^{FR} = \frac{\|G_{i+1}\|^2}{\|G_i\|^2}$$
 (Fletcher-Reeves) (24)

$$\gamma_i^{DY} = \frac{\|G_{i+1}\|^2}{\Re \operatorname{tr}(\tau H_i(G_{i+1} - \tau G_i))} \quad \text{(Dai-Yuan)}$$
$$\gamma_i^* = -\frac{\Re \operatorname{tr}(G_{i+1}(G_{i+1} - \tau G_i))}{\Re \operatorname{tr}(H_i G_i)}$$

Note, that although all of the above formulas are generalizations of the linear conjugate gradient method, the only formula that is motivated for inexact line search in the Euclidian case is the one of Hestenes-Stiefel, cf. [5]. It can also be shown that for our proposed step size (22), the term $\Re tr(\tau H_i G_{i+1})$ is negligible, leading to a motivation for γ_i^* .

We can use a result in [7] to show that using step size (22) together with the formula γ_i^* or γ_i^{HS} in the *Euclidian case* leads under some additional assumptions on the cost function - to N-step quadratic convergence. For a proof we refer to a forthcoming paper by the authors. We therefore have reason to assume, that this also holds in the manifold case.

Conjecture: Let f be a smooth function on $Gr_{k,n}$ with nondegenerate maximum Z. If the CG-method with step-size selection (22) and direction update λ_i^* or λ^{HS} converges to Z, then this convergence is N-step quadratically fast (where $N = \dim Gr_{k,n}$), i.e.

$$||P_{i+N} - Z|| \le \text{constant} ||P_i - Z||^2$$

for all i large enough, where $(P_i)_{i\in\mathbb{N}}$ denotes the sequence generated by the algorithm.

4.3. Cost Function

Let $A = A^{\dagger}$. A suitable cost function for our purpose is

$$f: Gr_{m,n} \to \mathbb{R}, \quad P \mapsto \Re \operatorname{tr}(PA).$$
 (25)

Proposition 3. *The Riemannian gradient and the Riemannian Hessian operator of f are given by*

$$\nabla f(P) = [P, [P, A]], \quad \mathcal{H}_{f(P)}(F) = -[P, [A, F]],$$
 (26)

where $F \in T_P Gr_{m,n}$. Moreover,

(a) $P = \theta N \theta^{\dagger}$ is critical point of f if and only if $\tilde{A} = U^{\dagger} A U$ is blockdiagonal, i.e.

$$\tilde{A} = \begin{bmatrix} A_{11} & 0\\ 0 & A_{22} \end{bmatrix},$$

where A_{11} is Hermitian $(k \times k)$ and A_{22} is Hermitian $(n - k) \times (n - k)$.

(b) P is a maximum of f if and only if $\operatorname{tr} A_{11} = \sum_{i=1}^{m} \lambda_i$, where $\lambda_1 \geq \cdots \geq \lambda_n$ are the eigenvalues of A. This maximum is unique on $\operatorname{Gr}_{m,n}$ if $\lambda_m > \lambda_{m+1}$.

Proof. Details can be found in a forthcoming paper on Newton's method for Graßmannians by U. Helmke, K. Hüper and J. Trumpf.

4.4. Implementation

Let $St_{k,n} := \{ u \in \mathbb{C}^{n \times k} \mid u^{\dagger}u = I_k \}$. For an implementation we exploit the fact that

$$Gr_{k,n} = \{ uu^{\dagger} \mid u \in St_{k,n} \}.$$

Moreover, it is easily checked that for arbitrary $X \in \mathbb{C}^{n \times n}$ and $P \in Gr_{k,n}$ the identity

$$[P, [P, [P, X]]] = [P, X]$$
(27)

holds and hence

$$[P, [P, \xi]] = \xi, \quad \text{for all} \quad \xi \in T_P Gr_{k,n}.$$
(28)

Note, that since $P = uu^{\dagger}$, the commutator [P, X] = PX - XP can be computed in $O(n^2k)$ steps.

The proposed algorithm has order $O(n^2k)$. It is of local nature and provides good results if the columns of the initial matrix $u_0 \in$ $St_{k,n}$ are close to the principal components of the Hermitian matrix A.

Algorithm 1 (CG-sweep). For a given estimation $u_0 \in St_{k,n}$ of the principal components of a Hermitian matrix A, this algorithm overwrites u_0 with u_{l-1} such that u_{l-1} is a (much) better approximation of the principal components of A.

- Set $P_0 = u_0 u_0^{\dagger}$ and set $\Omega_0 = [P_0, A]$, $G_0 := [P_0, \Omega_0]$, $H_0 := -G_0$.
- For i = 0, ..., l 1:

(s1) Set
$$\Delta := (\Omega_i^2)_{upp}$$
, set

$$\gamma'(0) := \Re \operatorname{tr}(P_i A \Omega_i)$$

$$\gamma''(0) := \Re \operatorname{tr}(\Delta P_i A) - \Re \operatorname{tr}(\Omega P \Omega A)$$
(29)

and, according to Eq. (22) and Lemma 1,

$$\lambda_i := -\frac{\gamma'(0)}{\gamma''(0)}.$$
(30)

(s2) Let
$$Q := (I + \lambda_i \Omega)_Q$$
; update $u_{i+1} := Qu_i$ and
 $P_{i+1} = u_{i+1}u_{i+1}^{\dagger}$ and set
 $\tau G_i := QG_i Q^{\dagger}$
 $\tau \Omega_i := Q[H_i, P_i]Q^{\dagger}$
 $\tau H_i := [\tau \Omega_i, P_{i+1}]$
 $G_{i+1} := [P_{i+1}, [P_{i+1}, A]].$

(s3) Compute γ_i according to one of the formulas in Eq. (24) and set $H_{i+1} := -G_{i+1} + \gamma_i \tau H_i$ $\Omega_{i+1} = [H_{i+1}, P_{i+1}].$

Notice, that the computation of Q in step (s2) can be performed in $O(n^2k)$ floating point operations. We shortly sketch how this can be achieved. Since Ω is at most of rank 2k, an amount of (2k+1)n - (2k+1)(2k+2)/2 Givens rotations suffice to obtain Hessenberg form, for example by annihilating matrix entries column wise from bottom to one under the diagonal. These rotations only depend on the lower triangular entries of Ω and the same rotations compute a Hessenberg form of $I + \Omega$. A subsequent annihilation of the remaining 2k + 1 entries on the subdiagonal completes the QRcomputation. The multiplication of the Givens rotations then results in $O(n^2k + k^2n)$ flops. See also [3] for how to use Givens rotations for computing the QR-factorization.

5. NUMERICAL SIMULATIONS AND DISCUSSION

Several simulations have been performed for the different choices of γ_i in Eq. (24). The Hermitian matrix $A \in \mathbb{C}^{n \times n}$ has been chosen to be nearly blockdiagonal with the dominant eigenvalues in the upper block in order to assure that $u_0 = \begin{bmatrix} I_k \\ 0 \end{bmatrix}$ is a good approximation of the principal components. In the following, k was chosen to be k = 5, while 3 different choices for n have been implemented, namely $n_1 = 50, n_2 = 100, n_3 = 150$. For the simulations, the matrix u has been extended to an orthonormal basis $\theta = [u, v]$ of \mathbb{C}^n and the stopping criterion has been chosen to be

blockoff
$$(\theta^* A \theta) \le 10^{-20}$$
,

where

blockoff
$$(A) = \sum_{k < i \le n, \ 1 \le j < k} (A_{ij})^2.$$

The number of steps needed until the stopping criterion was fulfilled is given in Table 1 for the different choices of γ_i and n_i .

Table 1. Number of steps for the different choices of γ_i until blockoff $\leq 10^{-20}$. In all experiments, k = 5.

	γ_i^{HS}	γ_i^{PR}	γ_i^{FR}	γ_i^{DY}	γ_i^*
$n_1 = 50$	11	12	15	16	12
$n_2 = 100$	12	12	30	35	13
$n_3 = 150$	15	16	> 100	> 100	16

Our observations can be summarized as follows. The formula of Hestenes and Stiefel, Polak-Ribière and γ_i^* perform equally well, with a slight advantage on the Hestenes-Stiefel formula. These cases confirm the well known behavior of CG-methods, cf. [5], that only very few steps within *one* sweep are required to obtain satisfactory

results. Moreover, it seems that the number of required steps only increases very slowly with the size of the problem. The proposed manifold adaptions of the formulas of Fletcher-Reeves and Dai-Yuan, however, turn out not to be suitable for the considered problem.

Finally, we would like to mention that numerical simulations support global convergence of the algorithm, if the step-size (30) is slightly modified into

$$\tilde{\lambda}_i := \frac{\gamma'(0)}{|\gamma''(0)|}.\tag{31}$$

Note, that locally around a nondegenerate maximum of f we have $\gamma''(0) \leq -\delta < 0$. Hence locally, this modification does not change the algorithm.

6. REFERENCES

- R. Everson and S. Roberts. Inferring the Eigenvalues of Covariance Matrices from Limited, Noisy Data. *IEEE Trans. Signal Processing*, 48(7):2083–2091, 2000.
- [2] D. Gabay. Minimizing a Differentiable Function over a Differential Manifold. J. Optimization Theory and Appl., 37(2):177– 219, 1982.
- [3] G.H. Golub and C.F. van Loan. *Matrix Computations, 3rd Edition* The Johns Hopkins University Press, 1996.
- [4] U. Helmke and J.B. Moore. *Optimization and Dynamical Systems*. Springer, London, 1994.
- [5] J. Nocedal and S.J. Wright. *Numerical Optimization*. Springer Series in Operations Research, 1999.
- [6] S.T. Smith. Optimization Techniques on Riemannian Manifolds. AMS Fields Inst. Commun., 3:113–136, 1994.
- [7] J. Stoer. On the Relation Between Quadratic Termination and Convergence Properties of Minimization Algorithms. *Numer*. *Math.*, 28:343–366, 1977.
- [8] H. Wang and M. Kaveh. Coherent Signal-Subspace Processing for the Detection and Estimation of Angles of Arrival of Multiple Wide-Band Sources. *IEEE Trans. Acoust., Speech, Signal Processing* 33(4): 823–831, 1985.
- [9] X. Yang, T.K. Sarkar, and E. Arvas. A Survey of Conjugate Gradient Algorithms for Solution of Extreme Eigen-Problems of a Symmetric Matrix. *IEEE Trans. Acoust., Speech, Signal Processing* 37(10): 1550–1556, 1989.