Traian Abrudan, Jan Eriksson, and Visa Koivunen. 2005. Optimization under unitary matrix constraint using approximate matrix exponential. In: Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers (ACSSC 2005). Pacific Grove, CA, USA. 28 October - 1 November 2005, pages 242-246.

Optimization under Unitary Matrix Constraint using Approximate Matrix Exponential

Traian Abrudan, Jan Eriksson, Visa Koivunen
SMARAD CoE, Signal Processing Laboratory, Helsinki University of Technology, P.O.Box 3000, Espoo, FINLAND 02015
e-mail: {tabrudan, jamaer, visa}@wooster.hut.fi

***Abstract*— In many engineering applications we deal with constrained optimization problems w.r.t. complex valued matrices. This paper proposes a Riemannian geometry approach for optimization of a real valued cost function $\mathcal{J}$ of *complex* valued matrix argument W, under the constraint that W is an $n \times n$ unitary matrix. An approximate steepest descent algorithm based on Taylor series expansion is developed. The approximation satisfies the unitary matrix constraint accurately even if low order approximation is used. Armijo adaptive step size rule [1] is used while moving towards the optimum. In the simulation examples, the proposed algorithm is applied to array signal processing and communications problems. The method outperforms other widely used algorithms.**

I. Introduction

Constrained optimization problems arise in many applications. In particular, we are addressing the problem of optimization under unitary matrix constraint. Such problems may be found in communications and array signal processing, for example, blind and constrained beamforming, high-resolution direction finding (e.g. MUSIC and ESPRIT), and generally all subspace-based methods where subspace tracking is needed. In addition, this type of optimization problems occur in Multiple-Input Multiple-Output (MIMO) communication systems and blind signal separation. See, [2] for recent review. Typically in communications and signal processing applications we are dealing with complex matrices and signals. Consequently, the methods derived for real-valued signals and orthogonal matrices may not be applicable. The extension from real [3] to complex case and unitary matrices is not trivial. It is not obtained just by replacing the transposition operation by the Hermitian transposition and the real derivative by the complex derivative, respectively.

Commonly, a cost function with unitary matrix constraint is minimized in the space of $n \times n$ matrices using a classical Steepest Descent Algorithm (SDA) with separate orthogonalization step applied in each iteration [4], [5]. The method of Lagrange multipliers where deviations from the unitarity property are penalized has also been employed in such problems [6]. A major improvement over the classical methods above is proposed in [7]. This differential geometry based method performs the optimization under *unitary* matrix constraints in an appropriate parameter space.

In this paper we propose an algorithm stemming from differential geometry for optimization of a real-valued cost function $\mathcal{J} : \mathbb{C}^{n\times n} \rightarrow \mathbb{R}$ subject to $\mathbf{W}\mathbf{W}^H = \mathbf{W}^H\mathbf{W} = \mathbf{I}$. This constrained optimization problem in $\mathbb{C}^{n\times n}$ may be translated into an unconstrained one in a different parameter space, i.e., the Lie group of $n \times n$ unitary matrices $U(n)$. A steepest descent algorithm operating in such parameter space is proposed. The exact method proposed in our earlier work [8] performs geodesic motion, i.e., it moves along locally length-minimizing paths towards the optimum. It requires the computation of a matrix exponential which may be too expensive in certain applications. In order to reduce the computational cost, we propose an approximate method that uses truncated Taylor series expansion. Even a low order model may be used since the approximate method does not suffer from error propagation. Consequently, the unitary matrix constraint may be satisfied with high fidelity in adaptive algorithms.

This paper is organized as follows. In Section II we introduce the Riemannian gradient in the unitary group and a steepest descent algorithm employing Taylor series expansion is derived. Simulation results and array signal processing and communications applications are presented in Section III. Finally, Section IV concludes the paper.

II. Algorithm

In this section we propose a steepest descent algorithm operating in the Lie group of $n \times n$ unitary matrices $U(n)$. In order to reduce the computational cost, we propose an approximate alternative to the exact algorithm proposed in [8]. The exact algorithm optimizes constrained cost function $\mathcal{J}(\mathbf{W})$ along geodesics on $U(n)$. The Riemannian gradient of the cost function evaluated at $\mathbf{W}$ and translated to identity is given by:

$$\mathbf{G}(\mathbf{W}) \triangleq \mathbf{\Gamma}_{\mathbf{W}}\mathbf{W}^H - \mathbf{W}\mathbf{\Gamma}_{\mathbf{W}}^H, \quad \in T_{\mathbf{I}}U(n), \tag{1}$$

where $\mathbf{\Gamma}_{\mathbf{W}} = \nabla\mathcal{J}(\mathbf{W})$ is the gradient of $\mathcal{J}$ in the $\mathbf{R}^{2n\times 2n}$ Euclidean space [9] at a given $\mathbf{W}$. The cost function $\mathcal{J}(\mathbf{W})$ is minimized iteratively, and the geodesic motion may be described by using an exponential map:

$$\mathbf{W}_{k+1} = \mathrm{expm}(-\mu\mathbf{G}_k)\mathbf{W}_k \triangleq \mathbf{R}_k\mathbf{W}_k, \tag{2}$$

where $\mathbf{G}_k = \mathbf{G}(\mathbf{W}_k)$ (1), the parameter $\mu > 0$ controls the algorithm convergence speed and $\mathbf{R}_k$ is unitary (rotation) matrix. Note that *the argument of the matrix exponential operation is a skew-Hermitian matrix*. The equation (2) is *the exact update* from [8]). This exploits the fact that the unitary matrices form a Lie group under the multiplication operation.

[1]This work was supported by the Academy of Finland and GETA Graduate School.

Product of unitary matrices is a unitary matrix. Hence, the multiplicative update in (2) satisfies the constraint in each step. Finding the rotation matrix $\mathbf{R}_k$ requires the computation of the matrix exponential operation (*expm*). The *expm* operation may be too expensive in some applications.

We propose a low complexity approximate algorithm based on Taylor series expansion of the *expm* operation. The effect of the approximation order is studied. Approximate algorithms do not necessarily satisfy the unitary matrix constraint exactly. However, already a low order approximation (order 3 to 5) produces accurate results. The truncated Taylor series approximation of *expm* of order ω is

$$\mathrm{expm}(\mathbf{A}) \approx \sum_{m=0}^{\omega} \frac{\mathbf{A}^m}{m!}. \tag{3}$$

The corresponding approximate update is $\mathbf{W}_{k+1} = \tilde{\mathbf{R}}_k \mathbf{W}_k$, where $\tilde{\mathbf{R}}_k$ is the approximate rotation matrix. For example, the second order approximation is

$$\tilde{\mathbf{R}}_k = \mathbf{I} - \mu \mathbf{G}_k + \frac{\mu^2}{2} \mathbf{G}_k^2, \tag{4}$$

where μ is the step size. After expanding the expression of the Riemannian gradient $\mathbf{G}_k$, the corresponding update is:

$$\begin{aligned} \mathbf{W}_{k+1} = \mathbf{W}_k &- \mu \big[\mathbf{\Gamma}_k \mathbf{W}_k^H \mathbf{W}_k - \mathbf{W}_k \mathbf{\Gamma}_k^H \mathbf{W}_k \big] \\ &+ \frac{\mu^2}{2} \big[(\mathbf{\Gamma}_k \mathbf{W}_k^H)^2 \mathbf{W}_k + (\mathbf{W}_k \mathbf{\Gamma}_k^H)^2 \mathbf{W}_k \\ &- \mathbf{\Gamma}_k \mathbf{W}_k^H \mathbf{W}_k \mathbf{\Gamma}_k^H \mathbf{W}_k - \mathbf{W}_k \mathbf{\Gamma}_k^H \mathbf{\Gamma}_k \mathbf{W}_k^H \mathbf{W}_k \big]. \end{aligned} \tag{5}$$

The multiplicative update in (2) turns into an additive update. The error propagation is an important practical issue. Since the unitary constraint is satisfied only approximately, the weighting factor $\mathbf{W}_k^H \mathbf{W}_k$ in equation (5) is different from the identity matrix. This weighting factor affects directly the unitary property of $\mathbf{W}_{k+1}$. If the weighting factor is ignored in eq. (5) (i.e., if we assume $\mathbf{W}_k^H \mathbf{W}_k = \mathbf{I}$), then even more severe degradation in the performance and departure from the unitarity property is experienced. The weighting factor improves the accuracy of the update in terms of unitary criterion similarly to the self-stabilized gradient algorithms in [10]. One important advantage of the proposed approximate algorithm is that the deviation from unitarity remains constant after a number of iterations, and it does not accumulate error as it can be seen in simulation results. This is due to the fact that the gradient (1) is always skew-Hermitian. The remaining error depends on the truncation order of the Taylor series. For a small step size, the third order truncated Taylor series approximates the matrix exponential with high fidelity. This is shown in [11], and verified by our simulations.

The proposed approximation of the matrix exponential operation requires $(\omega-1)[n^3+2n^2]$ operations (multiplications and additions). This takes into account the the special skew-Hermitian structure of $\mathbf{G}_k$. The approximation requires computing matrix powers in (3) which may be done very efficiently for skew-Hermitian matrices. The odd powers of $\mathbf{G}_k$ are also skew-Hermitian and the even powers are Hermitian. Consequently, the complexity of computing matrix powers is reduced approximately by half. The GPD method in [12] used to compute the matrix exponential requires $10n^3$ operations for skew-Hermitian matrices. Hence, Taylor series approximation of order $\omega \leq 10$ is justified.

An optimal value of the step size μ is difficult to determine in practice, since the matrices involved in the cost function may be random. Moreover, they may be time-varying. An adaptive step size is a reliable choice. It is known that the steepest descent algorithm together with the Armijo rule [1] for choosing the step size almost always converges *to a local minimum* if not initialized at a stationary point.

The proposed algorithm is summarized in Table I using a third order Taylor series approximation of the matrix exponential. The step size μ evolves in a dyadic basis. If it is too small, it will be doubled, or if it is too high it will be halved. The criteria for choosing the step size value are defined by two inequalities, the steps **6** and **7**, respectively.

1	Initialization: $k = 0$,$\mathbf{W}_k = \mathbf{I}$ and $\mu = 1$
2	Compute the gradient of the cost function in the Euclidean space: $\mathbf{\Gamma}_k = \frac{\partial \mathcal{J}}{\partial \mathbf{W}^*}(\mathbf{W}_k)$
3	Compute the gradient direction in the Riemannian space: $\mathbf{G}_k = \mathbf{\Gamma}_k \mathbf{W}_k^H - \mathbf{W}_k \mathbf{\Gamma}_k^H$
4	Evaluate $\|\mathbf{G}_k\|^2_{\mathbf{W}_k} = \mathrm{trace}\{\mathbf{G}_k \mathbf{G}_k^H\}$. If it is sufficiently small, then STOP.
5	Determine the approximate rotation matrices: $\tilde{\mathbf{R}}_k = \mathbf{I} - \mu\mathbf{G}_k + (\mu\mathbf{G}_k)^2/2 - (\mu\mathbf{G}_k)^3/6, \quad \tilde{\mathbf{Q}}_k = \tilde{\mathbf{R}}_k \tilde{\mathbf{R}}_k$
6	While $\mathcal{J}(\mathbf{W}_k) - \mathcal{J}(\tilde{\mathbf{Q}}_k \mathbf{W}_k) \geq \mu \|\mathbf{G}_k\|^2_{\mathbf{W}_k}$ $\tilde{\mathbf{R}}_k := \tilde{\mathbf{Q}}_k, \quad \tilde{\mathbf{Q}}_k = \tilde{\mathbf{R}}_k \tilde{\mathbf{R}}_k, \quad \mu := 2\mu$
7	While $\mathcal{J}(\mathbf{W}_k) - \mathcal{J}(\tilde{\mathbf{R}}_k \mathbf{W}_k) < (\mu/2) \|\mathbf{G}_k\|^2_{\mathbf{W}_k}$ $\tilde{\mathbf{R}}_k = \mathbf{I} - \mu\mathbf{G}_k + (\mu\mathbf{G}_k)^2/2 - (\mu\mathbf{G}_k)^3/6, \quad \mu := \mu/2$
8	Update: $\mathbf{W}_{k+1} = \tilde{\mathbf{R}}_k \mathbf{W}_k$ and go to step **2**, $k := k+1$

TABLE I

THE PROPOSED APPROXIMATE ALGORITHM. THE TAYLOR SERIES APPROXIMATION ORDER IS $\omega = 3$

This type of step adaptation allows reducing the complexity. When the step size needs to be doubled (step **6**), the computation of the Taylor series approximation in not needed, because the rotation matrix $\tilde{\mathbf{Q}}_k \approx \mathrm{expm}(-2\mu\mathbf{G}_k)$, may be obtained by squaring the matrix $\tilde{\mathbf{R}}_k \approx \mathrm{expm}(-\mu\mathbf{G}_k)$. This it is a realistic assumption even though we deal with an approximation, because for normal matrices, the approximate *expm* computation via matrix squaring (when doubling the step size) *prevents the roundoff error accumulation* [11]. An Armijo type of update algorithm enjoys this benefit, since the argument of the matrix exponential is skew-Hermitian, i.e. a normal matrix. Larger step size causes larger approximation error in the Taylor series. This may be avoided by using the scaling and squaring approach (Method 3 from [11]).

III. APPLICATION EXAMPLES

In this section we test the proposed algorithm in two different examples of signal processing applications. The first application example is a subspace-based direction of arrival

(DOA) estimation used in smart antenna systems. The second one is a subspace method for (semi)blind channel estimation in MIMO OFDM systems.

1) Subspace-based direction of arrival estimation: The method requires the computation of the signal or noise subspace. Here, the subspaces are estimated using a diagonalization approach. The antenna array covariance matrix $\mathbf{\Sigma}$ is diagonalized by finding diagonal matrix $\mathbf{D} = \mathbf{W}^H\mathbf{\Sigma}\mathbf{W}$ such that $\mathbf{W}$ is a unitary matrix. The matrix $\mathbf{D}$ contains the eigenvalues of $\mathbf{\Sigma}$ and $\mathbf{W}$ is a unitary matrix whose columns are the eigenvectors of $\mathbf{\Sigma}$. They may be found iteratively by minimizing the off-diagonal elements of $\mathbf{W}^H\mathbf{\Sigma}\mathbf{W}$, w.r.t. $\mathbf{W}$, under the unitarity constraint on $\mathbf{W}$. This is equivalent to minimizing the following cost function:

$$\mathcal{J}(\mathbf{W}) = \|\mathbf{W}^H\mathbf{\Sigma}\mathbf{W} - \mathbf{I} \odot (\mathbf{W}^H\mathbf{\Sigma}\mathbf{W})\|_F^2, \qquad (6)$$

where $\odot$ denotes the elementwise matrix multiplication. The gradient of (6) is $\mathbf{\Gamma}_{\mathbf{W}} = 2\mathbf{\Sigma}\mathbf{W}\big[\mathbf{W}^H\mathbf{\Sigma}\mathbf{W} - \mathbf{I} \odot (\mathbf{W}^H\mathbf{\Sigma}\mathbf{W})\big]$.

In the first simulation the impact of the Taylor series approximation order ω is studied, for $\omega = 2,\ 3,\ 5$. The performance is studied in terms of convergence speed. Obviously, faster methods for finding the DoA are used in practice. Two figures of merit are considered: a *diagonality criterion* Δ and the *unitarity criterion* Ω. The diagonality criterion is defined as a ratio of two squared Frobenius norms, i.e., the one corresponding to the off-diagonal vs. the one corresponding to the diagonal elements of $\mathbf{D}$, in logarithmic scale, i.e., $\Delta = 10\lg[\|\mathrm{off}(\mathbf{D})\|_F^2/\|\mathrm{diag}(\mathbf{D})\|_F^2]$. The unitarity criterion is defined as the squared Frobenius norm of the deviation from the unitarity property also in a logarithmic scale, i.e., $\Omega = 10\lg\|\mathbf{W}\mathbf{W}^H - \mathbf{I}\|_F^2$. The results are depicted in Fig. 1.

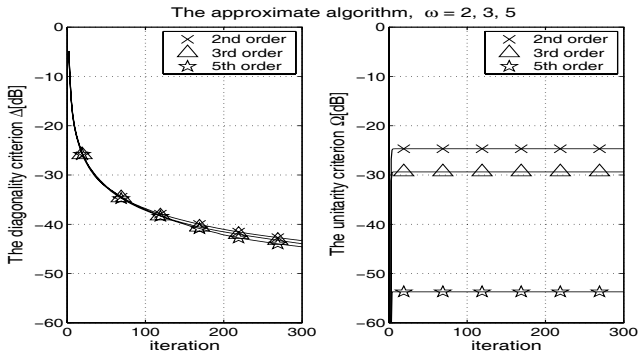

Fig. 1. The performance of the proposed algorithm (Table I) using the truncated Taylor series approximation of the matrix exponential operation. The diagonality criterion Δ (left) and the unitarity criterion Ω (right) vs. the iteration step. The unitarity criterion stabilizes to a steady-state value after few iterations. This value depends on the approximation order.

We may notice that approximation order does not impact the diagonality criterion significantly, see Fig. 1 (left). However, the approximation order plays a significant role in satisfying the unitarity property which is crucial in many applications, see, Fig. 1 (right).

In the second simulation we show the importance of unitarity and how it reflects on the DOA estimates. A 6-element uniform linear array (ULA) is used. The "true" DOAs are $\theta_1 = 87^o$ and $\theta_2 = 92^o$. The separation angle is 5^o, hence the high resolution property is needed. The obtained eigenvectors are plugged into standard MUSIC algorithm. We compare three algorithms: the classical steepest descent (SD) method which enforces unitarity in every iteration as in [4], a Lagrangian type of method as in [6], and the proposed approximate algorithm (Table I). We plot the spatial pseudo-

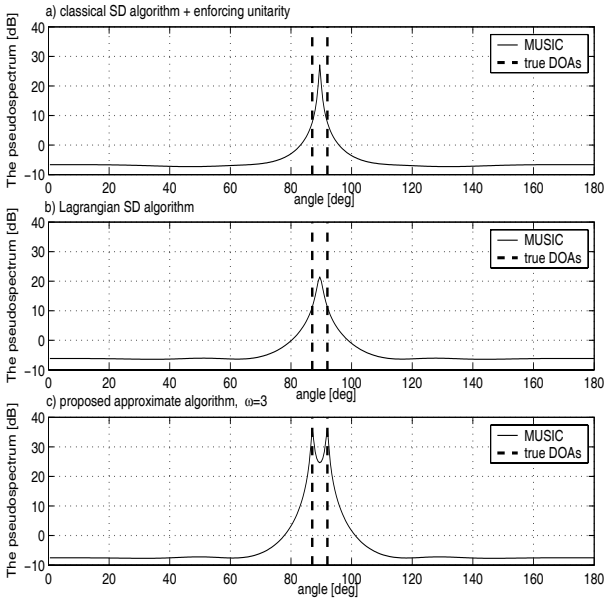

Fig. 2. The estimated DOAs (solid line) vs. the true DOAs: $\theta_1 = 87^o$ and $\theta_2 = 92^o$, (marked by dashed lines). The MUSIC algorithm is applied to a 6-element ULA at SNR=15dB. The eigendecomposition is obtained iteratively by minimizing the cost function (6), based on three different algorithms: a) The classical SD with enforcing unitarity every iteration b) The Lagrangian SD algorithm c) The proposed algorithm (Table I). The classical algorithms a), b) lose the high-resolution property. The proposed algorithm c) can solve closely-spaced DOA angles.

spectrum after 200 iterations for all methods. The classical SD algorithm produces the pseudo-spectrum in Fig. 2-a). It fails to detect both sources. This method converges to an accurate estimate only after several thousands of iterations. The Lagrangian method yields eigenvectors where the unitary property does not hold. Therefore, the orthogonality property of signal and noise subspaces is degraded and high resolution property is lost as well. See Fig. 2-b) where closely spaced sources remain unresolved. The proposed algorithm finds both signals as it is shown in Fig. 2-c). The peaks are high and well separated. This is obtained even if very poor initial estimates of the eigenvectors are used, i.e., the identity matrix. The method performs reliably in subspace-based estimation and tracking tasks where the unitarity property plays a crucial role.

2) Subspace method for semi-blind channel estimation in MIMO OFDM systems: In this example we consider the channel estimation algorithm for multi-user MIMO OFDM systems proposed in [13]. The same simulation parameters as in *Example 2* in [13] are considered. The MIMO system

has $K = 2$ transmit antennas and $J = 3$ receive antennas. All the MIMO channel branches are frequency selective and have order $L < 10$. The corresponding taps are zero-mean complex Gaussian, mutually independent generated according to the exponential power-delay profile $E[|h^{(j,k)}(l)|^2] = \exp(-0.64l)$, $l = 0, \ldots, L$. The OFDM block contains $N = 32$ sub-symbols belonging to 16-QAM constellation. Each block is zero-padded, resulting to a block of length $M = 41$.

The blind channel identification algorithm proposed in [13] is based on second order statistics of the received signal. The algorithm consists of the following steps. First, a sample estimate of the auto-correlation matrix $\hat{\mathbf{R}}_x$ is computed based on a finite number of received OFDM blocks. Ideally, the auto-correlation matrix is given by $\mathbf{R}_x = E[\mathbf{x}_i \mathbf{x}_i^H] = \mathbf{H}\mathbf{R}_u\mathbf{H}^H + \sigma_\nu^2 \mathbf{I}_{JM}$, where $\mathbf{x}_i$ is the receive antenna array output and $\mathbf{H}$ is the $JM \times KN$ block-Toeplitz matrix modeling the MIMO channel. $\mathbf{R}_u$ is the auto-correlation matrix of the transmitted signals and σ_ν^2 is the noise variance. The noise subspace is identified from the eigendecomposition (ED) of $\hat{\mathbf{R}}_x$. There are $q = JM - KN$ eigenvectors $\boldsymbol{\beta_i}$ corresponding to the noise subspace, therefore the classical subspace unitarity property $\boldsymbol{\beta_i}^H \mathbf{H} = \mathbf{0}$ holds. In the next step, each eigenvector $\boldsymbol{\beta_i}$ is re-arranged into a $N \times (L+1)J$ block-Toeplitz matrix $\mathbf{G}_i$, and and the resulting matrices are stacked into a large matrix $\mathbf{G}$. The MIMO channel coefficients are also reshaped in a $J(L+1) \times K$ matrix $\bar{\mathbf{H}}$. An equivalent unitarity equality $\mathbf{G}\bar{\mathbf{H}} = \mathbf{0}$ is obtained. The matrix $\bar{\mathbf{H}}$ may be determined up to a $K \times K$ ambiguity matrix $\mathbf{B}$, i.e., $\bar{\mathbf{H}} = \bar{\mathbf{H}}_0\mathbf{B}$. The matrix $\bar{\mathbf{H}}$ is a basis of the right null space of $\mathbf{G}$. This can be obtained from the singular value decomposition (SVD) of $\mathbf{G}$. Therefore, the columns of $\bar{\mathbf{H}}_0$ are formed by the right singular vectors corresponding to the K smallest singular values. The ambiguity matrix $\mathbf{B}$ may be removed based on pilot data, i.e., at least K sub-symbols in one OFDM block must be known. In conclusion, the channel estimation method proposed in [13] requires one ED of the auto-correlation matrix $\mathbf{R}_x$, followed by a SVD of the matrix $\mathbf{G}$ formed with the eigenvectors $\boldsymbol{\beta_i}$.

The proposed approximate algorithm reduces the complexity with a small performance degradation. First, we compare the complexity of the algorithm proposed in [13] by using the exact ED and SVD operations to the proposed iterative algorithm. An exact ED of the $JM \times JM$ matrix $\mathbf{R}_x$ requires about $13(JM)^3 \approx 24 \cdot 10^6$ operations including both additions and multiplications. The exact SVD of the $Nq \times (L+1)J$ matrix $\mathbf{G}$ requires about $2(Nq)[(L+1)J]^2 + 11[(L+1)J]^3 \approx 3.7 \cdot 10^6$ operations [14]. By using the proposed approximate algorithm (Table I) a complexity reduction may be achieved. We assume low approximation orders, i.e., $\omega_1 = 4$ for the ED of the $\mathbf{R}_x$ matrix and $\omega_2 = 5$ for the SVD of $\mathbf{G}$. The approximation of the ED operation reduces the complexity more than four times, i.e., about $(\omega_1 - 1)[(JM)^3 + 2(JM)^2] \approx 5.7 \cdot 10^6$ operations are required. Moreover, the SVD operation is converted to an ED operation. Instead of computing the right singular vectors of a large tall matrix $\mathbf{G}$ of size $Nq \times (L+1)J = 1288 \times 30$, we compute the eigenvectors of a smaller matrix $\mathbf{G}^H\mathbf{G}$ of size 30×30. In this way we further reduce the complexity. The multiplication $\mathbf{G}^H\mathbf{G}$ requires about $\frac{1}{2}(JM - KN)(L+1)[J^2N(L+2) - J(J-1)] + \frac{1}{2}[(L+1)J]^4 \approx 1.3 \cdot 10^6$ operations by exploiting the block-Toeplitz structure of $\mathbf{G}$. Finding the eigenvalues and the associated eigenvectors by using the proposed iterative approximation requires about $(\omega_2 - 1)[((L+1)J)^3 + 2((L+1)J)^2] \approx 0.1 \cdot 10^6$ operations. Therefore, the SVD may be approximated in $1.4 \cdot 10^6$ operations which is less than half of the complexity of the exact method.

We compare the exact and the approximated algorithms in order to evaluate the performance loss due to the approximation. In this simulation we perform both the ED and the SVD by using the proposed iterative approximation. The sample estimate of the auto-correlation matrix $\mathbf{R}_x$ is computed by using 200 received OFDM blocks. The number of iterations for the computation of the ED is also equal to 200, for both EDs.

After the MIMO channel is estimated, the equalization is performed in time domain and the ambiguity $\mathbf{B}$ due to the blind identification is removed as in [13]. The error is averaged over 50 Monte Carlo realizations. In Fig. 3 the performance of the proposed approximation is compared to the exact method [13] in terms of root mean-square error (RMSE) in the channel coefficients, as a function of the signal-to-noise ratio (SNR). Both the SNR and the channel RMSE are defined in [13]. The symbol RMSE is defined in the same manner for each user, and the average value over all users is considered. The performance degradation on the channel estimate due to the approximation may be noticed at high SNR. In terms of symbol RMSE the gap between the exact method [13] and the proposed approximate algorithm is not very significant, as it is shown in Fig. 4. The demodulated constellation patterns for the two users for the exact method and for the approximate method are shown in Fig. 5-a) and Fig. 5-b), respectively. The SNR is 21.4 dB, as in [13].

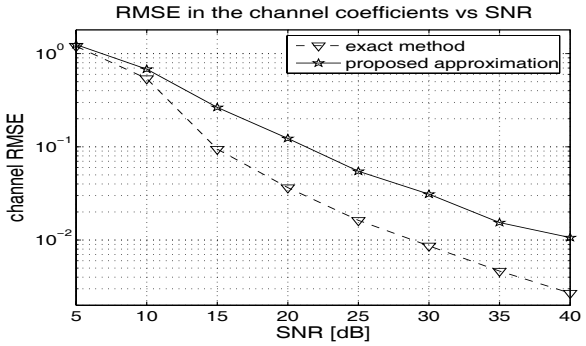

Fig. 3. The channel estimation RMSE as a function of SNR.

The algorithm convergence speed is also evaluated in terms of both channel RMSE and symbol RMSE. It may be noticed in Fig. 6 that after approximately 10 iterations both the channel and the symbol RMSE decrease significantly. This is due to the fact that after a certain number of iterations

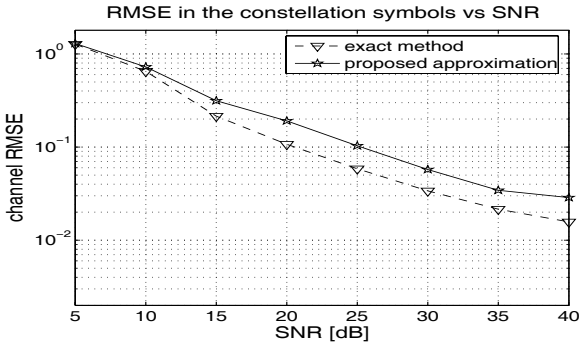

Fig. 4. The symbol RMSE as a function of SNR.

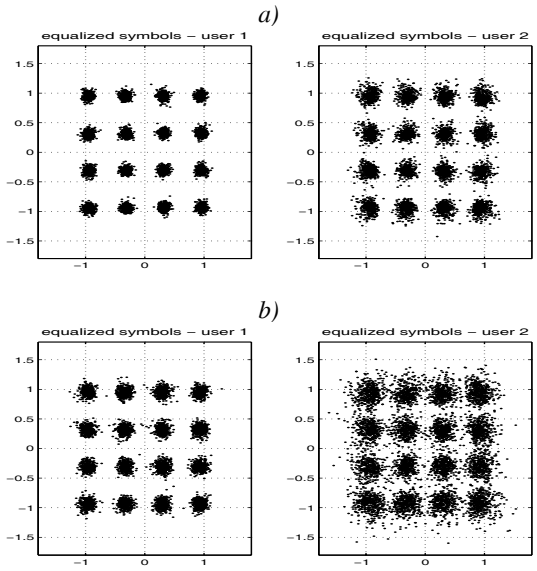

Fig. 5. The constellation patterns corresponding to the two users at SNR=21.4 dB. *a)* The exact method [13]. The channel RMSE is equal to 0.04 and the symbol RMSE is equal to 0.11. *b)* The proposed approximate algorithm from Table I. The channel RMSE is equal to 0.11 and the symbol RMSE is equal to 0.19.

the smallest eigenvalues may be distinguished easily and the proper eigenvectors are plugged in the algorithm.

IV. Conclusions

In this paper, an approximate Riemannian optimization algorithm under unitary matrix constraint is proposed. The truncated Taylor series approximation reduces the complexity and is very robust in the face of error propagation. Armijo step size [1] as well as more classical adaptation rules may be used in the update. The proposed method may be applied, for example, to smart antenna algorithms, wireless communications, biomedical measurements, signal separation, subspace estimation and tracking tasks where unitary matrices play an important role in general. Comparison to classical steepest descent and Lagrangian methods in given as well.

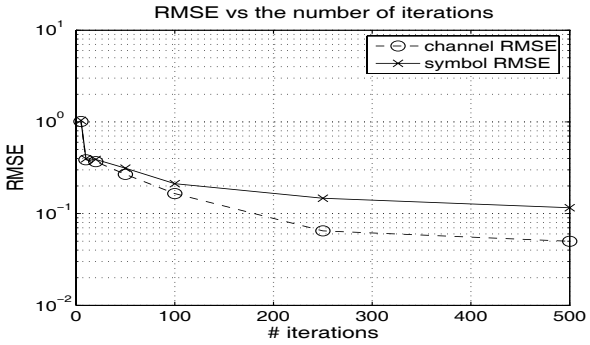

Fig. 6. The channel and the symbols RMSE vs. the number of iterations at SNR=21.4 dB.

The proposed algorithm provides significant advantages over the classical methods in terms of computational complexity. In terms of accuracy the approximation approaches the exact method.

References

[1] E. Polak, *Optimization: Algorithms and Consistent Approximations*. New York: Springer-Verlag, 1997.

[2] J. H. Manton, "On the role of differential geometry in signal processing," in *International Conference on Acoustics, Speech and Signal Processing*, vol. 5, pp. 1021–1024, Philadelphia, March 2005.

[3] S. Fiori, "Quasi-geodesic neural learning algorithms over the orthogonal group: a tutorial," *Journal of Machine Learning Research*, vol. 1, pp. 1–42, Apr. 2005.

[4] P. Sansrimahachai, D. Ward, and A. Constantinides, "Multiple-Input Multiple-Output Least-Squares Constant Modulus Algorithms," in *IEEE Global Telecommunications Conference*, vol. 4, pp. 2084–2088, 1–5 Dec. 2003.

[5] C. B. Papadias and A. M. Kuzminskiy, "Blind source separation with randomized Gram-Schmidt orthogonalization for short burst systems," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 809–812, 17–21 May 2004.

[6] L. Wang, J. Karhunen, and E. Oja, "A Bigradient optimization approach for robust PCA, MCA and source separation," in *Proceedings of IEEE Conference on Neural Networks*, vol. 4, pp. 1684–1689, 27 Nov.–1 Dec. 1995.

[7] J. H. Manton, "Optimization algorithms exploiting unitary constraints," *IEEE Transactions on Signal Processing*, vol. 50, pp. 635–650, Mar. 2002.

[8] T. Abrudan, J. Eriksson, and V. Koivunen, "Optimization under unitary matrix constaints via multiplicative update," *submitted to IEEE Transactions on Signal Processing*, May 2005.

[9] D. H. Brandwood, "A complex gradient operator and its applications in adaptive array theory," *IEE Proceedings, Parts F and H*, vol. 130, pp. 11–16, Feb. 1983.

[10] S. C. Douglas, "Self-stabilized gradient algorithms for blind source separation with orthogonality constraints," *IEEE Transactions on Neural Networks*, vol. 11, pp. 1490–1497, Nov. 2000.

[11] C. Moler and C. van Loan, "Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later," *SIAM Review*, vol. 45, no. 1, pp. 3–49, 2003.

[12] A. Iserles and A. Zanna, "Efficient computation of the matrix exponential by general polar decomposition," *SIAM Journal on Numerical Analysis*, vol. 42, pp. 2218–2256, March 2005.

[13] Y. Zeng and T. Ng, "A semi-blind channel estimation method for multiuser multiantenna OFDM systems," *IEEE Transactions on Signal Processing*, vol. 52, pp. 1419–1429, May 2004.

[14] G. H. Golub and C. van Loan, *Matrix computations*. The Johns Hopkins University Press, Baltimore, 3rd ed., 1996.