A Nested ℓ_1-penalized Adaptive Normalized Quasi-Newton Algorithm for Sparsity-Aware Generalized Eigen-subspace Extraction

Kengo Uchida† and Isao Yamada†
†Tokyo Institute of Technology, Japan
E-mail: {uchida, isao}@sp.ce.titech.ac.jp

***Abstract*—The sparsity-aware generalized eigen-subspace extraction is a modern strategy to achieve better interpretability than classical statistical data analysis, and has been realized, as sparse PCA, sparse CCA and sparse FDA, etc, in signal processing, machine learning and data sciences. For its broader applications in the scenarios of adaptive signal processing, the generalized orthogonality among the estimates of principal generalized eigenvectors is certainly desired to be exploited in the learning process. However, it seems that such adaptive learning algorithms have not yet been reported so far. In this paper, we present an algorithm by combining the idea of ℓ_1-penalized adaptive normalized quasi-Newton algorithm (Uchida and Yamada, 2018) with Nested orthogonal complement structure (NTY 2013, KYY 2017).**

I. Introduction

The generalized Hermitian eigenvalue problem (GHEP) is the problem of finding $\boldsymbol{w} \in \mathbb{C}^N \setminus \{\boldsymbol{0}\}$ and $\lambda \in \mathbb{R}$ such that

$$\boldsymbol{R_y w} = \lambda \boldsymbol{R_x w} \tag{1}$$

where $\boldsymbol{R_y}, \boldsymbol{R_x} \in \mathbb{C}^{N\times N}$ are Hermitian positive definite matrices, the scalar λ and the vector $\boldsymbol{w}$ are called respectively the generalized eigenvalue and generalized eigenvector corresponding to the matrix pencil $(\boldsymbol{R_y}, \boldsymbol{R_x})$. In a special case where $\boldsymbol{R_x}$ is the identity matrix $\boldsymbol{I}_N \in \mathbb{C}^{N\times N}$, the problem is simply referred to as the Hermitian eigenvalue problem (HEP). This matrix pencil has N positive generalized eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N\ (> 0)$ and their corresponding generalized eigenvectors $\boldsymbol{v}_i\ (i = 1, \cdots, N)$ can be chosen to satisfy $\boldsymbol{R_x}$-orthonormality, i.e.,

$$\begin{aligned} &\boldsymbol{R_y v}_i = \lambda_i \boldsymbol{R_x v}_i \\ \text{s.t.}\quad &\boldsymbol{v}_i^H \boldsymbol{R_x v}_j = \delta_{i,j} \quad (i, j = 1, 2, \cdots, N) \end{aligned} \tag{2}$$

where $\delta_{i,j}$ is the Kronecker delta function. The GHEP has many applications which includes Principal component analysis (PCA), Canonical correlation analysis (CCA) and Fisher discriminant analysis (FDA). These have been used extensively in wide range of data sciences, e.g., signal processing, machine learning, pattern recognition, control engineering, etc. We should remark that the principal generalized eigenvectors $\boldsymbol{v}_i$ of GHEP can also serve as the principal component of the transformed observed data via tight frame [1], [2], e.g., Wavelet transform.

In the scenarios of adaptive subspace tracking (see, e.g., [3]–[7]), the matrix pencil $(\boldsymbol{R_y}, \boldsymbol{R_x})$ correspond respectively to covariance matrices of given observed random input sequences $(\boldsymbol{y}_{(k)})_{k\in\mathbb{N}}$,$(\boldsymbol{x}_{(k)})_{k\in\mathbb{N}}$ and are not available at time k. In this case, the principal generalized eigenvectors $\boldsymbol{v}_i$ must be estimated at time k as $\boldsymbol{w}_{(k)}$ from a certain estimated sequence $(\boldsymbol{R}_{\boldsymbol{y}(l)}, \boldsymbol{R}_{\boldsymbol{x}(l)})_{0\leq l\leq k}$ of $(\boldsymbol{R_y}, \boldsymbol{R_x})$. Such adaptive estimation of $\boldsymbol{v}_i$ is required and some effective adaptive estimators are proposed (see, e.g., [5]–[7]).

On the other hand, especially in data science analyses, despite the simplicity and popularity of these analyses, they have one potential difficulty in the actual interpretation of their results. For instance, in PCA, the principal components have an actual physical meaning in many applications. Therefore, for better interpretability of result, the sparsity of eigenvector (suitable modifications of eigenvector) have been desired, and algorithms for such sparsity-aware eigenvector analysis have been reported, e.g., sparse PCA [8]–[12] and sparse generalized eigenvalue problem (sparse GEP [13]–[15]). However, in [12], it is pointed out that, although the advantages of an orthogonal basis are well known, the orthogonality property of the estimates is sacrificed for sparse solution in many of the sparse PCA algorithms. Note that it is almost clear that a generalized orthogonality (i.e., $\boldsymbol{R_x}$-orthogonality) among the estimates of principal generalized eigenvectors is important in sparse GEP.

Based on the above background, we proposed an adaptive algorithm to exploit the sparsity as a priori knowledge in the estimation of the first principal generalized eigenvector $\boldsymbol{v}_1$ (introduced in [16]). In this paper, aiming to exploiting the sparsity in the estimation of $\boldsymbol{v}_i$ without trading off their $\boldsymbol{R_x}$-orthogonality, we present multidimensional extension of the idea of [16]. In fact, we first reduce the estimation problem of $\boldsymbol{v}_i(i = 1, \cdots, r)$ of $(\boldsymbol{R_y}, \boldsymbol{R_x})$ to that of the first principal generalized eigenvector $\boldsymbol{v}_1^{(i)} \in \mathbb{C}^{N-i+1}$ of a certain smaller matrix pencil $(\boldsymbol{R}_{\boldsymbol{y}}^{(i)}, \boldsymbol{R}_{\boldsymbol{x}}^{(i)})(i = 1, \cdots, r)$ by using the nested orthogonal complement structure [17], [18]. To enhance the sparsity of $\boldsymbol{v}_i$ in the original domain, we introduce a sparsity promoting penalty to a non-convex criterion employed in [7]. The proposed algorithm is naturally in the frame of the normalized quasi-Newton's strategy in a similar idea to [16]. In numerical experiments, we demonstrate that the following two points: (i)The proposed algorithm has excellent tracking performance on sparsity-aware generalized eigen-subspace extraction model. (ii)The proposed algorithm can also keep the $\boldsymbol{R_x}$-orthogonality in adaptive estimation scenario.

II. Preliminaries

A. A Summary of Nested Orthogonal Complement Structure

Let $(\boldsymbol{A},\boldsymbol{B})$ be an $N\times N$ Hermitian positive definite matrix pencil. Denote its generalized eigenvalues by $\mu_i(\mu_1\le\mu_2\le\cdots\le\mu_N)$ and their corresponding generalized eigenvectors by $\boldsymbol{u}_i\in\mathbb{C}^N\backslash\{\boldsymbol{0}\}$. In [17], a scheme is proposed for tracking the first r principal (or minor) generalized eigenvectors with keeping the orthogonality. This scheme is designed by the dimension reduction technique introduced in [19] for the case $\boldsymbol{B}=\boldsymbol{I}_N$, where $\boldsymbol{I}_N\in\mathbb{R}^{N\times N}$ is the identity matrix. The key of the dimension reduction technique for generalized eigenvector extraction is the $\boldsymbol{B}$-orthogonal complement matrix defined as follows.

Definition 1. [$\boldsymbol{B}$-orthogonal complement matrix]
Let $\boldsymbol{B}\in\mathbb{C}^{N\times N}$ be a Hermitian positive definite matrix. For $\boldsymbol{u}\in\mathbb{C}\backslash\{\boldsymbol{0}\},\boldsymbol{U}_{\perp[\boldsymbol{B}]}\in\mathbb{C}^{N\times(N-1)}$ is called a $\boldsymbol{B}$-orthogonal complement matrices of $\boldsymbol{u}$ if

$$(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}))^H\boldsymbol{B}\boldsymbol{u}=\boldsymbol{0}\text{ and }(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}))^H(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}))=\boldsymbol{I}_{N-1}. \tag{3}$$

Fact 1. [Example of a $\boldsymbol{B}$-orthogonal complement matrix]
One of $\boldsymbol{B}$-orthogonal complement matrices of $\boldsymbol{u}$ can be calculated as

$$\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}):=\begin{pmatrix}\boldsymbol{I}_{N-1}-\frac{1}{1+|\overline{u}_{low}|}\overline{\boldsymbol{u}}_{up}\overline{\boldsymbol{u}}_{up}^H\\ -\theta(\overline{u}_{low})\overline{\boldsymbol{u}}_{up}^H\end{pmatrix} \tag{4}$$

where $\overline{\boldsymbol{u}}_{up}\in\mathbb{C}^{N-1}$ and $\overline{u}_{low}\in\mathbb{C}$ are respectively the first $N-1$ components and the last component of a normalized vector $\overline{\boldsymbol{u}}:=\boldsymbol{B}\boldsymbol{u}/\|\boldsymbol{B}\boldsymbol{u}\|$, i.e., $\overline{\boldsymbol{u}}=(\overline{\boldsymbol{u}}_{up},\overline{u}_{low})^T$, and $\theta:\mathbb{C}\to\mathbb{C}$ is defined as

$$\theta(\overline{u}_{low}):=\begin{cases}1 & \text{if } \overline{u}_{low}=0,\\ \overline{u}_{low}/|\overline{u}_{low}| & \text{otherwise.}\end{cases} \tag{5}$$

Fact 2. [Expression of non-first minor generalized eigenvectors]
(a) Let $\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1)$ be the B-orthogonal complement matrix of $\boldsymbol{u}_1\in\mathbb{C}$ and define

$$\begin{aligned}\hat{\boldsymbol{A}}&=(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1))^H\boldsymbol{A}(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1))\in\mathbb{C}^{(N-1)\times(N-1)},\\ \hat{\boldsymbol{B}}&=(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1))^H\boldsymbol{B}(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1))\in\mathbb{C}^{(N-1)\times(N-1)}.\end{aligned} \tag{6}$$

Then, $(\hat{\boldsymbol{A}},\hat{\boldsymbol{B}})$ is a Hermitian positive matrix pencil whose generalized eigenvalues are given by $(0<)\mu_2\le\mu_3\le\cdots\le\mu_N$. Moreover, $\boldsymbol{u}_i(i=2,\cdots,N)$ can be expressed as

$$\boldsymbol{u}_i=(\boldsymbol{U}_{\perp[\boldsymbol{B}]}(\boldsymbol{u}_1))\hat{\boldsymbol{u}}_{i-1} \tag{7}$$

where $\hat{\boldsymbol{u}}_{i-1}\in\mathbb{C}^{N-1}$ is the $(i-1)$th generalized eigenvector corresponding to μ_i of matrix pencil $(\hat{\boldsymbol{A}},\hat{\boldsymbol{B}})$ and satisfies $\hat{\boldsymbol{u}}_{i-1}^H\hat{\boldsymbol{B}}\hat{\boldsymbol{u}}_{j-1}=\delta_{i,j}$.

(b) (Nested orthogonal complement structure)

Define N matrix pencils $(\boldsymbol{A}^{(i)},\boldsymbol{B}^{(i)})$ recursively as $(\boldsymbol{A}^{(1)},\boldsymbol{B}^{(1)}):=(\boldsymbol{A},\boldsymbol{B})$ and

$$\begin{aligned}\boldsymbol{A}^{(i+1)}&:=(\boldsymbol{U}_\perp^{(i)})^H\boldsymbol{A}^{(i)}(\boldsymbol{U}_\perp^{(i)})\in\mathbb{C}^{(N-i)\times(N-i)},\\ \boldsymbol{B}^{(i+1)}&:=(\boldsymbol{U}_\perp^{(i)})^H\boldsymbol{B}^{(i)}(\boldsymbol{U}_\perp^{(i)})\in\mathbb{C}^{(N-i)\times(N-i)},\end{aligned} \tag{8}$$

where

$$\boldsymbol{U}_\perp^{(i)}:=\boldsymbol{U}_{\perp[\boldsymbol{B}^{(i)}]}(\boldsymbol{u}_1^{(i)})\in\mathbb{C}^{(N-i+1)\times(N-1)} \tag{9}$$

and $\boldsymbol{u}_1^{(i)}\in\mathbb{C}^{N-i+1}$ is the first minor generalized eigenvector of $(\boldsymbol{A}^{(i)},\boldsymbol{B}^{(i)})$. Then, $(\boldsymbol{A}^{(i)},\boldsymbol{B}^{(i)})$ are Hermitian positive definite matrix pencils whose generalized eigenvalues are given by $(0<)\mu_i\le\mu_{i+1}\le\cdots\le\mu_N$. The ith minor generalized eigenvector $\boldsymbol{u}_i^{(1)}(=\boldsymbol{u}_i)$ of $(\boldsymbol{A},\boldsymbol{B})(i=2,\cdots,N)$ is expressed as

$$\boldsymbol{u}_i^{(1)}=\boldsymbol{U}_\perp^{(1)}\cdots\boldsymbol{U}_\perp^{(i-1)}\boldsymbol{u}_1^{(i)}=\perp_i\boldsymbol{u}_1^{(i)} \tag{10}$$

where

$$\perp_i:=\prod_{s=1}^{i-1}\boldsymbol{U}_\perp^{(s)}. \tag{11}$$

By combining Fact1,2 and any iterative algorithm for computing the first minor generalized eigenvector of the matrix pencil $(\boldsymbol{A},\boldsymbol{B})$, we can estimate the first r minor generalized eigenvectors in sequence. Note that the smallest generalized eigenvalue of $(\boldsymbol{A},\boldsymbol{B})$ is the inverse of the largest generalized eigenvalue of the matrix pencil $(\boldsymbol{B},\boldsymbol{A})$. Moreover, the minor generalized eigenvector corresponding to the smallest generalized eigenvalue of $(\boldsymbol{A},\boldsymbol{B})$ is the principal generalized eigenvector corresponding to the largest generalized eigenvalue of $(\boldsymbol{B},\boldsymbol{A})$.

B. Adaptive Normalized Quasi-Newton Algorithm (ANQNA)

We present a summary of the generalized eigen-pair estimator in [7]. This algorithm estimates a stationary point $(\overline{\boldsymbol{w}},\overline{\lambda})\in\mathbb{C}^N\times\mathbb{R}$ of the following function ξ with Newton's strategy:

$$\xi(\boldsymbol{w},\lambda):=\boldsymbol{w}^H\boldsymbol{R_y}\boldsymbol{w}\lambda^{-1}-\boldsymbol{w}^H\boldsymbol{R_x}\boldsymbol{w}+\ln\lambda. \tag{12}$$

The stationary point of ξ is given as a zero point following equation:

$$\begin{pmatrix}\frac{\partial\xi}{\partial\boldsymbol{w}}\\ \frac{\partial\xi}{\partial\lambda}\end{pmatrix}=\begin{pmatrix}2\boldsymbol{R_y}\boldsymbol{w}\lambda^{-1}-2\boldsymbol{R_x}\boldsymbol{w}\\ -\boldsymbol{w}^H\boldsymbol{R_y}\boldsymbol{w}\lambda^{-2}+\lambda^{-1}\end{pmatrix}. \tag{13}$$

Hence, the stationary point $(\overline{\boldsymbol{w}},\overline{\lambda})$ satisfies

$$\begin{cases}\boldsymbol{R_y}\overline{\boldsymbol{w}}=\overline{\lambda}\boldsymbol{R_x}\overline{\boldsymbol{w}}\\ \overline{\boldsymbol{w}}^H\boldsymbol{R_y}\overline{\boldsymbol{w}}=\overline{\lambda}\end{cases}. \tag{14}$$

This means that $(\overline{\boldsymbol{w}},\overline{\lambda})$ is a generalized eigen-pair of the matrix pencil $(\boldsymbol{R_y},\boldsymbol{R_x})$. When the stationary point of ξ is obtained by using Newton's method, the update formula can be written as:

$$\begin{pmatrix}\tilde{\boldsymbol{w}}_{(k+1)}\\ \lambda_{(k+1)}\end{pmatrix}=\begin{pmatrix}\boldsymbol{w}_{(k)}\\ \lambda(k)\end{pmatrix}-\eta\boldsymbol{H}[\xi]^{-1}(\boldsymbol{w}_{(k)},\lambda_{(k)})\nabla\xi(\boldsymbol{w}_{(k)},\lambda_{(k)}) \tag{15}$$

where $\eta > 0$ is the step size parameter. At this time, we need to calculate the gradient $\nabla\xi$ and the inverse of the Hessian matrix $\boldsymbol{H}[\xi]^{-1}$ at each time. However, ξ is a non-convex function, so it can't be said that $\boldsymbol{H}[\xi]$ is a regular matrix. Even if $\boldsymbol{H}[\xi]$ has its inverse at $[\boldsymbol{w}_{(k)}, \lambda_{(k)}] \in \mathbb{C}^N \times \mathbb{R}$, the computational complexity for the inversion of this Hessian matrix is $\mathcal{O}(N^3)$. Then, it is not practical for many online applications. In [7], it has been shown that the inverse of this can be approximated as follows:

$$\begin{aligned}\tilde{\boldsymbol{H}}_p^{-1}(\boldsymbol{w}, \lambda) &= \frac{1}{2}\begin{pmatrix} \frac{1}{2}\boldsymbol{w}\boldsymbol{w}^H - \boldsymbol{R}_{\boldsymbol{x}}^{-1} & -\boldsymbol{w}\lambda \\ -\boldsymbol{w}^H\lambda & 0 \end{pmatrix} \\ &\approx \begin{pmatrix} \frac{\partial^2\xi}{\partial^2\boldsymbol{w}} & \frac{\partial^2\xi}{\partial\boldsymbol{w}\partial\lambda} \\ \frac{\partial^2\xi}{\partial\lambda\partial\boldsymbol{w}} & \frac{\partial^2\xi}{\partial^2\lambda} \end{pmatrix}^{-1} = \boldsymbol{H}[\xi]^{-1}(\boldsymbol{w}, \lambda)\end{aligned} \tag{16}$$

in the estimation of the largest generalized eigenvalue λ_1 and the principal generalized eigenvector $\boldsymbol{v}_1$. Therefore, we can acceptably calculate the quasi-Newton step at each time in acceptable cost. By combining the following normalization step

$$\boldsymbol{w}_{(k+1)} = \frac{\tilde{\boldsymbol{w}}_{(k+1)}}{\|\tilde{\boldsymbol{w}}_{(k+1)}\|_{\boldsymbol{R}_{\boldsymbol{x}}}} \tag{17}$$

with the above quasi-Newton step (15) (16), an effective algorithm for GHEP (see [7, Algorithm1]). Moreover, in [7], the effectiveness of its adaptive implementation, with replacement of $(\boldsymbol{R}_{\boldsymbol{y}(k)}, \boldsymbol{R}_{\boldsymbol{x}(k)})$ by $(\boldsymbol{R}_{\boldsymbol{y}}, \boldsymbol{R}_{\boldsymbol{x}})$ at time k, has been discussed in the scenarios of subspace tracking.

C. ℓ_1-penalized adaptive normalized quasi-Newton algorithm

In [16], we have reported an ℓ_1-penalized extension of the Adaptive normalized quasi-Newton algorithm (ℓ_1-penalized ANQNA) which aims to exploit effectively the sparsity as a priori knowledge for efficient subspace tracking and is also motivated by recent sparsity-aware eigenvector analysis in data sciences. We have newly designed an objective function by adding to ξ in (12) ℓ_1-norm as a sparsity promoting penalty ψ, i.e.,

$$\zeta(\boldsymbol{w}, \lambda) := \xi(\boldsymbol{w}, \lambda) + \psi(\boldsymbol{w}), \tag{18}$$

where

$$\psi(\boldsymbol{w}) := -\rho\sum_{i=1}^{N}|w_i| \tag{19}$$

where $\rho \geq 0$. Note that ψ is nonpositive valued in (19). This is because ξ in (12) is locally maximized in the estimation of $\boldsymbol{v}_1$ in [7]. In order to derive a quasi-Newton type algorithm for locally maximizing ζ, the ℓ_1-penalized ANQNA is derived by applying a quasi-Newton step which an approximation of the gradient and inverse of Hessian matrix are used to the criterion ζ followed by normalization step. By using the ℓ_1-penalized ANQNA, we can obtain a certain modified solution in estimation of $\boldsymbol{v}_1$. Our proposed algorithm is based on this idea, so the details are described in the next section.

III. Nested ℓ_1-Penalized Adaptive Normalized Quasi-Newton Algorithm

In this section, we propose a Nested ℓ_1-penalized adaptive normalized quasi-Newton algorithm (Algorithm 1) to exploit sparsity in generalized eigen-subspace extraction. For simplicity, we focus on the pair of real symmetric positive matrices throughout this paper.

From the dimension reduction idea of Nested orthogonal compliment structure and (12), we define a non-convex optimization criterion $\xi^{(i)}$ of which stationary point corresponds to the generalized eigen-pair of $(\boldsymbol{R}_{\boldsymbol{y}}^{(i)}, \boldsymbol{R}_{\boldsymbol{x}}^{(i)}) \in \mathbb{R}^{(N-i+1)\times(N-i+1)}$ as

$$\xi(\boldsymbol{w}^{(i)}, \lambda) := (\boldsymbol{w}^{(i)})^T\boldsymbol{R}_{\boldsymbol{y}}^{(i)}\boldsymbol{w}^{(i)}\lambda^{-1} - (\boldsymbol{w}^{(i)})^T\boldsymbol{R}_{\boldsymbol{x}}^{(i)}\boldsymbol{w}^{(i)} + \ln\lambda. \tag{23}$$

Algorithm 1 can be derived with a extension similar to (18) and (19) by modifying an objective function $\xi^{(i)}$ as

$$\zeta^{(i)}(\boldsymbol{w}^{(i)}, \lambda) := \xi^{(i)}(\boldsymbol{w}^{(i)}, \lambda) + \psi^{(i)}(\boldsymbol{w}^{(i)}) \tag{24}$$

with ℓ_1 norm as a sparsity promoting penalty:

$$\psi^{(i)}(\boldsymbol{w}^{(i)}) := -\rho\| \perp_i \boldsymbol{w}^{(i)}\|_1 \tag{25}$$

where $\rho \geq 0$. A natural extension of the normalized quasi-Newton's strategy, for finding the stationary point of ζ would be

$$\begin{cases} \begin{pmatrix} \tilde{\boldsymbol{w}}_{(k+1)}^{(i)} \\ \lambda_{(k+1)} \end{pmatrix} = \begin{pmatrix} \boldsymbol{w}_{(k)}^{(i)} \\ \lambda_{(k)} \end{pmatrix} - \eta\tilde{\boldsymbol{H}}[\zeta]^{-1}(\boldsymbol{w}_{(k)}^{(i)}, \lambda_{(k)})\tilde{\nabla}\zeta(\boldsymbol{w}_{(k)}^{(i)}, \lambda_{(k)}) \\ \boldsymbol{w}_{(k+1)}^{(i)} = \dfrac{\tilde{\boldsymbol{w}}_{(k+1)}^{(i)}}{\|\tilde{\boldsymbol{w}}_{(k+1)}^{(i)}\|_{\boldsymbol{R}_{\boldsymbol{x}}}}, \end{cases} \tag{26}$$

where $\tilde{\nabla}\zeta(\boldsymbol{w}^{(i)}, \lambda)$ and $\tilde{\boldsymbol{H}}^{-1}[\zeta](\boldsymbol{w}^{(i)}, \lambda)$ are the approximations of the gradient of $\zeta^{(i)}$ and the inverse of Hessian of $\zeta^{(i)}$, respectively, $\eta > 0$ is a step size parameter. Let $\boldsymbol{C}^{(i)}$ be

$$\perp_i =: \boldsymbol{C}^{(i)} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_{N-i+1}]^T \in \mathbb{R}^{N\times(N-i+1)}. \tag{27}$$

Then, as the approximation of the gradient of $\zeta^{(i)}$, we use a subgradient of $\zeta^{(i)}$ given by

$$\tilde{\nabla}\zeta^{(i)} := \begin{pmatrix} 2\boldsymbol{R}_{\boldsymbol{y}}^{(i)}\boldsymbol{w}^{(i)}\lambda^{-1} - 2\boldsymbol{R}_{\boldsymbol{x}}^{(i)}\boldsymbol{w}^{(i)} - \rho\boldsymbol{g}^{(i)} \\ -(\boldsymbol{w}^{(i)})^T\boldsymbol{R}_{\boldsymbol{y}}\boldsymbol{w}^{(i)}\lambda^{-2} + \lambda^{-1} \end{pmatrix} \tag{28}$$

where $\boldsymbol{g}^{(i)} := (g_1, g_2, \cdots, g_{N-i})^T$ and $g_l = \sum_{j=1}^{N-i+1}\mathrm{sgn}(\boldsymbol{c}_j\boldsymbol{w}^{(i)})c_{jl}$. As a fair approximation of $\boldsymbol{H}[\zeta^{(i)}]^{-1}(\boldsymbol{w}^{(i)}, \lambda)$ at $(\boldsymbol{w}^{(i)}, \lambda) \approx (\boldsymbol{v}_1^{(i)}, \lambda_1^{(i)})$, from [7], we can use

$$\tilde{\boldsymbol{H}}[\zeta^{(i)}]^{-1}(\boldsymbol{w}^{(i)}, \lambda) := \frac{1}{2}\begin{pmatrix} \frac{1}{2}\boldsymbol{w}^{(i)}(\boldsymbol{w}^{(i)})^T - \boldsymbol{R}_{\boldsymbol{x}}^{-1} & -\boldsymbol{w}^{(i)}\lambda \\ -(\boldsymbol{w}^{(i)})^T\lambda & 0 \end{pmatrix} \tag{29}$$

where $(\boldsymbol{v}_1^{(i)}, \lambda_1^{(i)})$ is the generalized eigen-pair of $(\boldsymbol{R}_{\boldsymbol{y}}^{(i)}, \boldsymbol{R}_{\boldsymbol{x}}^{(i)})$. Hence, it is not hard to see that Algorithm 1 for the case of $\eta = \gamma$ can be derived by using this quasi-Newton type update (26), (28) and (29).

Algorithm 1 : NESTED ℓ_1-PENALIZED and NORMALIZED QUASI-NEWTON ALGORITHM

With $\boldsymbol{R}_{\boldsymbol{x}}^{(i)}$-normalized vector $\boldsymbol{w}_{(0)}^{(i)} \in \mathbb{R}^{(N-i+1)}$ and $\lambda_{(0)} > 0$, generate the sequence $(\boldsymbol{w}_{(k)}^{(i)}, \lambda_{(k)})(k = 0, 1, ...)$ by

$$\tilde{\boldsymbol{w}}_{(k+1)}^{(i)} := \boldsymbol{w}_{(k)}^{(i)} + \frac{\eta}{\lambda_{(k)}}\left\{(\boldsymbol{R}_{\boldsymbol{x}}^{(i)})^{-1}\boldsymbol{R}_{\boldsymbol{y}}^{(i)}\boldsymbol{w}_{(k)}^{(i)} - (\boldsymbol{w}_{(k)}^{(i)})^T\boldsymbol{R}_{\boldsymbol{y}}^{(i)}\boldsymbol{w}_{(k)}^{(i)}\boldsymbol{w}_{(k)}^{(i)} + \frac{\rho_{(k)}}{4}\left[\boldsymbol{w}_{(k)}^{(i)}(\boldsymbol{w}_{(k)}^{(i)})^T - 2(\boldsymbol{R}_{\boldsymbol{x}}^{(i)})^{-1}\right]\boldsymbol{g}_{(k)}^{(i)}\lambda_{(k)}\right\} \quad (20)$$

$$\boldsymbol{w}_{(k+1)}^{(i)} := \frac{\tilde{\boldsymbol{w}}_{(k+1)}^{(i)}}{\|\tilde{\boldsymbol{w}}_{(k+1)}^{(i)}\|_{\boldsymbol{R}_{\boldsymbol{x}}^{(i)}}} \quad (21)$$

$$\lambda_{(k+1)} := (1-\gamma)\lambda_{(k)} + \gamma(\boldsymbol{w}_{(k+1)}^{(i)})^T\boldsymbol{R}_{\boldsymbol{y}}^{(i)}\boldsymbol{w}_{(k+1)}^{(i)} - \frac{\rho_{(k)}\gamma}{2}(\boldsymbol{w}_{(k+1)}^{(i)})^T\boldsymbol{g}_{(k)}^{(i)}\lambda_{(k)} \quad (22)$$

where $g_l = \sum_{j=1}^{N-i+1}\operatorname{sgn}(\boldsymbol{c}_j\boldsymbol{w})c_{jl}$, $\eta > 0$ and $\gamma \in (0,1)$ are the step sizes, $\rho_{(k)} \geq 0$ is sparsity inducting parameter, and $(\boldsymbol{w}_{(k)}, \lambda_{(k)})$ are the estimates of the first sparse principal generalized eigenvector and eigenvalue.

By combining Nested orthogonal complement structure strategy, we can extend the idea of the ℓ_1-penalized ANQNA [16] to sparsity-aware generalized eigen-subspace extraction without trading off their orthogonality. We establish the following scheme (Scheme 1) for estimation of the first r sparse principal generalized eigenvectors and also present an adaptive version of Scheme 1 as Scheme 2.[1]

Scheme 1. [Extraction of the first r sparse generalized eigenvectors of the matrix pencil $(\boldsymbol{A}, \boldsymbol{B})$]

1) *Set* $\boldsymbol{A}^{(1)} = \boldsymbol{B}$ *and* $\boldsymbol{B}^{(1)} = \boldsymbol{A}$.
2) *For* $i = 1, \cdots, r$,
   a) *Extract the first principal generalized eigenvector* $\boldsymbol{u}_1^{(1)}$ *with* $(\boldsymbol{A}^{(i)}, \boldsymbol{B}^{(i)})$ *Algorithm1*
   b) *If* $i \neq r$, *compute the* $\boldsymbol{B}^{(i)}$*-orthogonal complement matrix* $\boldsymbol{U}_\perp^{(i)}$ *of* $\boldsymbol{u}_1^{(i)}$.
   c) *If* $i \neq r$, *set* $\boldsymbol{A}^{(i+1)} := (\boldsymbol{U}_\perp^{(i)})^T\boldsymbol{A}^{(i)}(\boldsymbol{U}_\perp^{(i)})$, $\boldsymbol{B}^{(i+1)} := (\boldsymbol{U}_\perp^{(i)})^T\boldsymbol{B}^{(i)}(\boldsymbol{U}_\perp^{(i)})$.
3) *For* $i = 2, \cdots, r$, *compute* $\boldsymbol{u}_i^{(1)}$ *by (10), i.e.,* $\boldsymbol{u}_i^{(1)} = \perp_i \boldsymbol{u}_1^{(i)}$.

Scheme 2. [Adaptive implementation of Scheme 1]

1) $k \leftarrow k+1$. *Update the estimate* $(\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}^{(1)}, \tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(1)})$ *of* $(\boldsymbol{R}_{\boldsymbol{y}}, \boldsymbol{R}_{\boldsymbol{x}})$
2) *For* $i = 1, \cdots, r$,
   a) *Perform only on update of Algorithm1 from* $(\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}^{(i)}, \tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i)})$ *and* $\boldsymbol{w}_{1(k-1)}^{(i)}$ *and denote the outcome by* $\boldsymbol{w}_{1(k)}^{(i)}$.
   b) *If* $i \neq r$, *compute the* $\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i)}$*-orthogonal complement matrix* $\boldsymbol{W}_{\perp(k)}^{(i)}$ *of* $\boldsymbol{w}_{1(k)}^{(i)}$.[2]
   c) *If* $i \neq r$, *set* $\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}^{(i+1)} := (\boldsymbol{W}_{\perp(k)}^{(i)})^T\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}^{(i)}(\boldsymbol{W}_{\perp(k)}^{(i)})$, $\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i+1)} := (\boldsymbol{W}_{\perp(k)}^{(i)})^T\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i)}(\boldsymbol{W}_{\perp(k)}^{(i)})$.
3) *For* $i = 2, \cdots, r$, *compute* $\boldsymbol{w}_{i(k)}^{(1)} = \left(\prod_{s=1}^{i-1}\boldsymbol{W}_{\perp(k)}^{(s)}\right)\boldsymbol{w}_{1(k)}^{(i)}$.
4) *Repeat steps 1-4 with* $\boldsymbol{w}_{i(k)}^{(1)} (i = 1, \cdots, r)$ *converge.*

IV. Numerical Experiments

In this section, we demonstrate numerically the performance of the proposed algorithm. In this experiment, as performance criterion, we use the $\boldsymbol{R}_{\boldsymbol{x}}$-direction cosine between $\boldsymbol{w}_{(k)}$ and $\boldsymbol{v}$:

$$DC_{\boldsymbol{R}_{\boldsymbol{x}}}(\boldsymbol{w}_{(k)}, \boldsymbol{v}) := \frac{|\langle\boldsymbol{w}_{(k)}, \boldsymbol{v}\rangle_{\boldsymbol{R}_{\boldsymbol{x}}}|}{\|\boldsymbol{w}_{(k)}\|_{\boldsymbol{R}_{\boldsymbol{x}}}\|\boldsymbol{v}\|_{\boldsymbol{R}_{\boldsymbol{x}}}}. \quad (34)$$

[1] In adaptive estimation, the matrix $\boldsymbol{R}_{\boldsymbol{y}}$ is unknown and has to be estimated. In many cases, $\boldsymbol{R}_{\boldsymbol{y}}$ is the covariance matrix of random input sequence $\{\boldsymbol{y}_{(k)}\}_{\mathbb{N}}$. In this paper, the sample covariance matrices $\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}$ is one-rank updated with the recursions [7], [20]:

$$\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k+1)} = \beta\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)} + \boldsymbol{y}_{(k+1)}\boldsymbol{y}_{(k+1)}^T, \quad (30)$$

$$\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k+1)} = \alpha\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)} + \boldsymbol{x}_{(k+1)}\boldsymbol{x}_{(k+1)}^T \quad (31)$$

where $\alpha, \beta \in (0,1)$. In this case, by using the matrix inversion lemma [21] and [17, Corollary 2], we can obtain the inversion of $\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(i)} := (\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i)})^{-1}$ as:

$$\boldsymbol{Q}_{\boldsymbol{x}(k+1)}^{(1)} = \frac{1}{\alpha}\left[\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(1)} - \frac{\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(1)}\boldsymbol{x}_{(k+1)}\boldsymbol{x}_{(k+1)}^T\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(1)}}{\alpha + \boldsymbol{x}_{(k+1)}^T\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(1)}\boldsymbol{x}_{(k+1)}}\right], \quad (32)$$

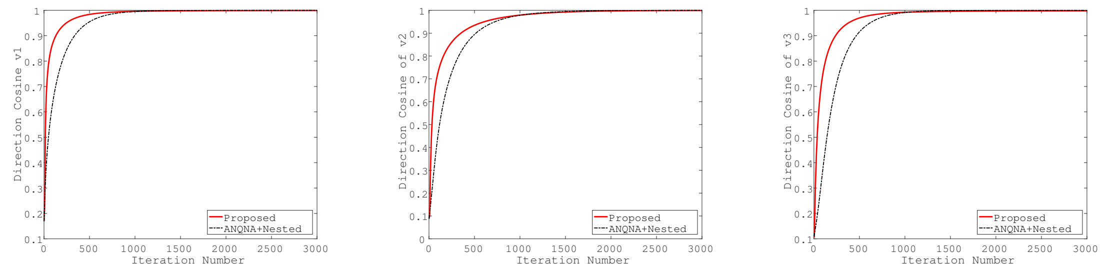
$$\boldsymbol{Q}_{\boldsymbol{x}(k+1)}^{(i+1)} = (\boldsymbol{U}_\perp^{(i)})^T\boldsymbol{Q}_{\boldsymbol{x}(k+1)}(\boldsymbol{U}_\perp^{(i)}) - (\boldsymbol{U}_\perp^{(i)})^T\boldsymbol{u}_1^{(i)}(\boldsymbol{u}_1^{(i)})^T(\boldsymbol{U}_\perp^{(i)}). \quad (33)$$

By replacing $\boldsymbol{R}_{\boldsymbol{y}}^{(i)}$, $\boldsymbol{R}_{\boldsymbol{x}}^{(i)}$ and $(\boldsymbol{R}_{\boldsymbol{x}}^{(i)})^{-1}$ in Algorithm 1 with $\tilde{\boldsymbol{R}}_{\boldsymbol{y}(k)}^{(i)}$, $\tilde{\boldsymbol{R}}_{\boldsymbol{x}(k)}^{(i)}$ and $\boldsymbol{Q}_{\boldsymbol{x}(k)}^{(i)}$, we can derive adaptive implementation.

[2] $\overline{\boldsymbol{w}}_{(k)}^{(i)} = \frac{\boldsymbol{R}_{\boldsymbol{x}(k)}^{(i)}\boldsymbol{w}_{1(k)}^{(i)}}{\|\boldsymbol{R}_{\boldsymbol{x}(k)}^{(i)}\boldsymbol{w}_{1(k)}^{(i)}\|}$,

$$\boldsymbol{W}_{\perp(k)}^{(i)} = \begin{pmatrix} \boldsymbol{I}_{N-i} - \frac{1}{1+|\overline{w}_{low(k)}^{(i)}|}\overline{\boldsymbol{w}}_{up(k)}^{(i)}\overline{\boldsymbol{w}}_{up(k)}^{(i)T} \\ -\theta(\overline{w}_{low(k)}^{(i)})\overline{\boldsymbol{w}}_{up(k)}^{(i)T} \end{pmatrix},$$

where $\overline{\boldsymbol{w}}_{(k)}^{(i)} = [\overline{\boldsymbol{w}}_{up(k)}^{(i)T}, \overline{w}_{low(k)}^{(i)}]$, and the mapping θ is defined as (5).



(a) Direction cosine in the estimation of $\boldsymbol{v}_1$ (b) Direction cosine in the estimation of $\boldsymbol{v}_2$ (c) Direction cosine in the estimation of $\boldsymbol{v}_3$

Fig. 1: Comparison of the performance of the proposed algorithm with the ANQNA (Section IV-A)

For precision, we observe Average $\boldsymbol{R_x}$-direction cosine of (34) over L independent runs,

$$ADC_{(k)} := \frac{1}{r}\sum_{i=1}^{r} DC_{\boldsymbol{R_x}}(\boldsymbol{w}_{i(k)}, \boldsymbol{v}_i). \tag{35}$$

We also measure the numerical stability of algorithms by the sample standard deviation of the average direction cosine (35):

$$SSD_{(k)} := \sqrt{\frac{1}{L-1}\sum_{j=1}^{L}[ADC_{j(k)} - \overline{ADC}_{(k)}]^2}, \tag{36}$$

where $ADC_{j(k)}$ is the $\boldsymbol{R_x}$-direction cosine of the jth independent run $(j = 1, \cdots, L)$ and $\overline{ADC}_{(k)}$ is the average of $L(= 100)$ independent runs (i.e., $\overline{ADC}_{(k)} := \frac{1}{L}\sum_{j=1}^{L} ADC_{j(k)}$). Moreover, in order to evaluate the orthogonality of the estimates, we observe the orthogonality error [5], [6], [17]:

$$\Theta_{(k)} := \|\boldsymbol{W}_{(k)}^T \boldsymbol{R}_{\boldsymbol{x}(k)} \boldsymbol{W}_{(k)} - \boldsymbol{I}_r\|_F^2 \tag{37}$$

where $\boldsymbol{W}_{(k)} := [\boldsymbol{w}_{1(k)}, \boldsymbol{w}_{2(k)}, \cdots, \boldsymbol{w}_{r(k)}]$, and $\|\cdot\|$ stands for the Frobenius norm.

A. Sparsity-aware subspace extraction

We compare the performance of the proposed scheme with that of combination of the Adaptive normalized quasi-Newton algorithm (ANQNA) [7] and Nested orthogonal complement structure [17], in the situation where the covariance matrix pencil of the input data has sparse principal generalized eigenvectors. We refer to the procedure proposed in [10], [14] to generate random data with a covariance matrix pencil having sparse eigenvector. We generate a positive definite symmetric matrix pancil $(\boldsymbol{C_y}, \boldsymbol{C_x})$ as $\boldsymbol{C_y} = \boldsymbol{V}^{-T}\mathrm{diag}(\boldsymbol{d})\boldsymbol{V}^{-1}$, $\boldsymbol{C_x} = \boldsymbol{V}^{-T}\boldsymbol{V}^{-1}$, where $\boldsymbol{V} \in \mathbb{R}^{N\times N}$ has pre-specified sparse vectors and the remaining columns are generated randomly. Here, we choose $N = 100$, where the two sparse generalized eigenvectors are specified as follows:

$$\begin{aligned}
\boldsymbol{v}_1 &:= \begin{cases} V_{i,1} = \frac{1}{\sqrt{5}} & \text{for } i = 1, ..., 5, \\ V_{i,1} = 0 & \text{otherwise,} \end{cases} \\
\boldsymbol{v}_2 &:= \begin{cases} V_{i,2} = \frac{1}{\sqrt{5}} & \text{for } i = 6, ..., 10, \\ V_{i,2} = 0 & \text{otherwise,} \end{cases} \\
\boldsymbol{v}_3 &:= \begin{cases} V_{i,3} = \frac{1}{\sqrt{5}} & \text{for } i = 11, ..., 15, \\ V_{i,3} = 0 & \text{otherwise,} \end{cases}
\end{aligned} \tag{38}$$

and the generalized eigenvalues are chosen as $d_1 = 40, d_2 = 20, d_3 = 10, d_i = 1 (i \geq 3)$.

In this subsection, the step size parameters is chosen as $\eta = \lambda_N/(\lambda_1 - \lambda_N)$ and $\gamma = 0.5$ in both algorithms, and the sparsity inducting parameter of the proposed algorithm is decreased as $\rho_{(k)} = \frac{1}{\sqrt{k}}$ (In the estimation of each $\boldsymbol{v}_i(i = 1, \cdots, 3)$, these parameter setting are all the same). $\boldsymbol{w}_{(0)}$ and $\lambda_{(0)}$ are randomly selected so as to satisfy $\|\boldsymbol{w}_{(0)}\|_{\boldsymbol{R_x}} = 1$ and $\lambda_{(0)} \in (0, 1)$. In Fig.1(a)(b)(c), we observe values and changes of direction cosine between the estimated eigenvector $\boldsymbol{w}_{(k)}$ and the sparse eigenvectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ and $\boldsymbol{v}_3$. Fig.1 shows that the proposed scheme can achieve faster tracking than the existing scheme by exploiting the sparsity as a priori knowledge.

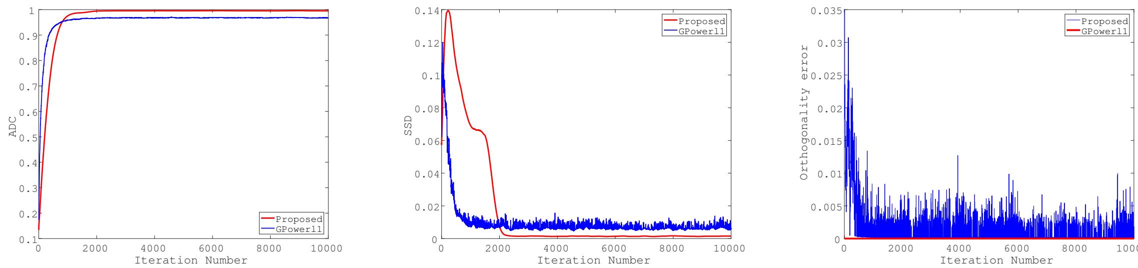
B. Potential application (sparse principal component analysis and its online version)

In this subsection, we can also see that it is found that the proposed algorithm can provide a more stable estimate on the *online sparse PCA* setting (a covariance matrix changes with input at time k).[3] As the existing method to compare, we adopt Generalized Power Method (GPower) [11] which is one of the most popular method for sparse PCA. In [11], two algorithms are proposed as methods to estimate single principal component with sparsity, named GPower$_{\ell_1}$ and GPower$_{\ell_0}$. In this experiments, we choose GPower$_{\ell_1}$ as benchmark algorithm because these GPower algorithms have very similar performance in terms of (34), (36) and (37).[4] We construct a covariance matrix with the eigenvalue decomposition $\boldsymbol{\Sigma} = \boldsymbol{V}\mathrm{diag}(\boldsymbol{d})\boldsymbol{V}^T$, where $\boldsymbol{V} \in \mathbb{R}^{N\times N}$ has several pre-specified sparse orthonormal vectors, where $\boldsymbol{V}$ has $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ and $\boldsymbol{v}_3$ in (38) and the remaining eigenvectors is defined to make its column vectors an orthonormal basis. We choose N = 50 in this subsection. In addition, we generate a input data sequence $\{\boldsymbol{y}_{(k)}\}_{k\in\mathbb{N}}$ from a zero-mean normal distribution with $\boldsymbol{\Sigma}$, i.e., $\boldsymbol{y}_{(k)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma})$ and add noise to input for a SNR of -5dB.

In Fig.2, we observe values and changes of averaged direction cosine between the estimated eigenvector $\boldsymbol{w}_{(k)}$ and the

[3]This setting suggests that we apply the proposed algorithm to a time-varying covariance matrix pencil $(\boldsymbol{R}_{\boldsymbol{y}(k)}, \boldsymbol{I}_N)$ which is updated with the recursion (30) and input data sequences $\{\boldsymbol{y}_{(k)}\}_{k\in\mathbb{N}}$ in this experimants.

[4]The GPower algorithm has been proposed as a batch type algorithm. In this experiments, we use its straightforward adaptive implementation with (30) as the existing algorithm to make a comparison in online condition.



(a) Average direction cosine of the estimates (b) Sample standard deviation of Fig.2(a) (c) Orthogonality error of the estimates

Fig. 2: Comparison of the performance of the proposed algorithm with the GPower algorithm (adaptive case, Section IV-B)

sparse eigenvectors $\boldsymbol{v}_1$, $\boldsymbol{v}_2$ and $\boldsymbol{v}_3$, and Fig.2(b) shows the sample standard deviation of the direction cosine (36). Moreover, Fig.2(c) shows the orthogonality error (37) of the estimates, for each time k. In this experiments, the step size parameters is chosen as $\eta = 0.01$ and $\gamma = 0.25$ in this subsection. The sparsity inducting parameter is fixed as $\rho^{(k)} = 0.05$. Note that all algorithms have been initialized with $\boldsymbol{R}_{\boldsymbol{y}(0)} = \boldsymbol{I}_N$. $\boldsymbol{w}_{(0)}$ and $\lambda_{(0)}$ are randomly selected so as to satisfy $\|\boldsymbol{w}_{(0)}\|_{\boldsymbol{R}_{\boldsymbol{x}}} = 1$ and $\lambda_{(0)} \in (0, 1)$ in this subsection. We set the forgetting factor as $\beta = 0.998$ in (30). GPower$_{\ell_1}$ needs to set sparsity-controlling parameters $\gamma_{\text{GP}1}, \cdots, \gamma_{\text{GP}3}$ and weight parameters $\mu_{\text{GP}1}, \cdots, \mu_{\text{GP}3}$ (respectively introduced as γ and μ in [11]). We choose them as $\gamma_{\text{GP}i} = 0.1$ $(i = 1, \cdots, 3)$ and $\mu_{\text{GP}i} = 1/i$. Fig.2(a) and Fig.2(b) suggests that the proposed algorithm can provide more stable convergence and steady state performance than the GPower in this online condition. From Fig.2(c), we can see that the estimates by the proposed algorithm keep the orthogonality in this online estimation.

V. Conclusion

In this paper, we have presented the Nested ℓ_1-penalized adaptive normalized quasi-Newton algorithm for sparsity-aware generalized eigen-subspace extraction. In fact, we have proposed an idea to extend our precious work on the algorithm for the first sparse principal generalized eigenvector [16] by adopting the Nested orthogonal compliment structure. Additionally, by imposing a sparsity-enhancing penalty function, the proposed algorithm can achieve the sparse i-th generalized eigenvector tracking for generalized eigenvalue problem without sacrificing a general orthogonality (i.e., $\boldsymbol{R}_{\boldsymbol{x}}$-orthogonality) among the solution. Numerical experiments have shown the following two points: (i) The proposed algorithm can exploit the sparsity in the case that the generalized eigenvectors have sparse property. (ii) In adaptive case as well as batch case, the proposed algorithm achieve keeping the orthogonality among the estimates of the principal eigenvectors.

References

[1] S.-C. Pei and M.-H. Yeh, "An introduction to discrete finite frames," *IEEE Signal processing magazine*, vol. 14, no. 6, pp. 84–96, 1997.
[2] S. Mallat, *A wavelet tour of signal processing*. Academic press, 1999.
[3] B. Yang, "Projection approximation subspace tracking," *IEEE Transactions on Signal processing*, vol. 43, no. 1, pp. 95–107, 1995.
[4] J. Yang, H. Xi, F. Yang, and Y. Zhao, "RLS-based adaptive algorithms for generalized eigen-decomposition," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1177–1188, 2006.
[5] S. Attallah and K. Abed-Meraim, "A fast adaptive algorithm for the generalized symmetric eigenvalue problem," *IEEE Signal Processing Letters*, vol. 15, pp. 797–800, 2008.
[6] T. Tanaka, "Fast generalized eigenvector tracking based on the power method," *IEEE Signal Processing Letters*, vol. 16, no. 11, pp. 969–972, 2009.
[7] T. D. Nguyen and I. Yamada, "Adaptive normalized quasi-Newton algorithms for extraction of generalized eigen-pairs and their convergence analysis," *IEEE Transactions on Signal Processing*, vol. 61, no. 6, pp. 1404–1418, 2013.
[8] I. T. Jolliffe, N. T. Trendafilov, and M. Uddin, "A modified principal component technique based on the lasso," *Journal of computational and Graphical Statistics*, vol. 12, no. 3, pp. 531–547, 2003.
[9] H. Zou, T. Hastie, and R. Tibshirani, "Sparse principal component analysis," *Journal of computational and graphical statistics*, vol. 15, no. 2, pp. 265–286, 2006.
[10] H. Shen and J. Z. Huang, "Sparse principal component analysis via regularized low rank matrix approximation," *Journal of multivariate analysis*, vol. 99, no. 6, pp. 1015–1034, 2008.
[11] M. Journée, Y. Nesterov, P. Richtárik, and R. Sepulchre, "Generalized power method for sparse principal component analysis," *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 517–553, 2010.
[12] K. Benidis, Y. Sun, P. Babu, and D. P. Palomar, "Orthogonal sparse pca and covariance estimation via procrustes reformulation," *IEEE Transactions on Signal Processing*, vol. 64, no. 23, pp. 6211–6226, 2016.
[13] B. K. Sriperumbudur, D. A. Torres, and G. R. Lanckriet, "A majorization-minimization approach to the sparse generalized eigenvalue problem," *Machine learning*, vol. 85, no. 1-2, pp. 3–39, 2011.
[14] J. Song, P. Babu, and D. P. Palomar, "Sparse generalized eigenvalue problem via smooth optimization," *IEEE Transactions on Signal Processing*, vol. 63, no. 7, pp. 1627–1642, 2015.
[15] S. E. Safo, J. Ahn, Y. Jeon, and S. Jung, "Sparse generalized eigenvalue problem with application to canonical correlation analysis for integrative analysis of methylation and gene expression data," *Biometrics*, 2016.
[16] K. Uchida and I. Yamada, "An ℓ_1-penalization of adaptive normalized quasi-Newton algorithm for sparsity-aware generalized eigenvector estimation," in *Statistical Signal Processing Workshop (SSP), 2018 IEEE International Conference on*. IEEE, 2018.
[17] T. D. Nguyen, N. Takahashi, and I. Yamada, "An adaptive extraction of generalized eigensubspace by using exact nested orthogonal complement structure," *Multidimensional Systems and Signal Processing*, pp. 1–27, 2013.
[18] K. Kakimoto, M. Yamagishi, and I. Yamada, "Smoothing of adaptive eigenvector extraction in nested orthogonal complement structure with minimum disturbance principle," *Multidimensional Systems and Signal Processing*, vol. 29, no. 1, pp. 433–465, 2018.
[19] M. Misono and I. Yamada, "An efficient adaptive minor subspace extraction using exact nested orthogonal complement structure," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 91, no. 8, pp. 1867–1874, 2008.
[20] S. Haykin, "Adaptive filter theory," *2nd.ed., Prentice-Hall, Englewood Cliffs, NJ*, 1991.
[21] W. W. Hager, "Updating the inverse of a matrix," *SIAM review*, vol. 31, no. 2, pp. 221–239, 1989.