A Decomposition Algorithm for the Sparse Generalized Eigenvalue Problem

Ganzhao Yuan[1,3,4], Li Shen[2], Wei-Shi Zheng[3,4]

[1] Center for Quantum Computing, Peng Cheng Laboratory, Shenzhen 518005, China
[2] Tencent AI Lab, Shenzhen, China   [3] School of Data and Computer Science, Sun Yat-sen University, China
[4] Key Laboratory of Machine Intelligence and Advanced Computing (Sun Yat-sen University), Ministry of Education, China

yuanganzhao@foxmail.com, mathshenli@gmail.com, wszheng@ieee.org

Abstract

The sparse generalized eigenvalue problem arises in a number of standard and modern statistical learning models, including sparse principal component analysis, sparse Fisher discriminant analysis, and sparse canonical correlation analysis. However, this problem is difficult to solve since it is NP-hard. In this paper, we consider a new effective decomposition method to tackle this problem. Specifically, we use random or/and swapping strategies to find a working set and perform global combinatorial search over the small subset of variables. We consider a bisection search method and a coordinate descent method for solving the quadratic fractional programming subproblem. In addition, we provide some theoretical analysis for the proposed method. Our experiments on synthetic data and real-world data have shown that our method significantly and consistently outperforms existing solutions in term of accuracy.

1. Introduction

In this paper, we mainly focus on the following sparse generalized eigenvalue problem ('$\triangleq$' means define):

$$\min_{\mathbf{x}\neq\mathbf{0},\mathbf{x}\in\Omega} f(\mathbf{x}) \triangleq \frac{h(\mathbf{x})}{g(\mathbf{x})}, \text{ with } \Omega \triangleq \{\mathbf{x} \mid \|\mathbf{x}\|_0 \leq s\}, \quad (1)$$
$$h(\mathbf{x}) \triangleq \tfrac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x}, \ \ g(\mathbf{x}) \triangleq \tfrac{1}{2}\mathbf{x}^T\mathbf{C}\mathbf{x}.$$

Here, $\mathbf{x} \in \mathbb{R}^n$, and $\|\cdot\|_0$ is a function that counts the number of nonzero elements in a vector. $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $\mathbf{C} \in \mathbb{R}^{n\times n}$ are some symmetry matrices. We assume that $\mathbf{C}$ is strictly positive definite and $s \in [1, n]$ is a positive integer.

The sparse generalized eigenvalue problem in (1) describes many applications of interest in both computer vision and machine learning, including object recognition [26], visual tracking [21], object detection [27, 28, 31], pixel/part selection [25], and text summarization [44]. We notice that the objective function and sparsity constraint in (1) is scale-invariant (multiplying $\mathbf{x}$ with a positive constant does not change the value of the objective function and the satisfiability of the sparsity constraint). Thus, it is equivalent to the following problem: $\min_{\mathbf{x}} \ \mathbf{x}^T\mathbf{A}\mathbf{x}$, $s.t.$ $\mathbf{x}^T\mathbf{C}\mathbf{x} = 1$, $\|\mathbf{x}\|_0 \leq s$. Moreover, without the sparsity constraint, Problem (1) reduces to the minimum generalized eigenvalue problem and it has several equivalent formulations [4]: $\min_{\mathbf{x}\neq\mathbf{0}} \ (\mathbf{x}^T\mathbf{A}\mathbf{x})/(\mathbf{x}^T\mathbf{C}\mathbf{x}) = \min\{\mathbf{x}^T\mathbf{A}\mathbf{x} : \mathbf{x}^T\mathbf{C}\mathbf{x} = 1\} = \max\{\lambda : \mathbf{A} - \lambda\mathbf{C} \succeq \mathbf{0}\} = \lambda_{\min}(\mathbf{C}^{-1/2}\mathbf{A}\mathbf{C}^{-1/2})$, where $\lambda_{\min}(\mathbf{X})$ is the smallest eigenvalue of a given matrix $\mathbf{X}$, and $\mathbf{X} \succeq \mathbf{0}$ denotes $\mathbf{X}$ is positive semidefinite.

Problem (1) is closely related to the classical matrix computation in the literature [13, 11, 1]. Imposing an additional sparsity constraint on the solution reduces over-fitting and improves the interpretability of the model for high-dimensional data analysis. The work of [18] successively choose a sparse principle direction that maximizes the variance by enforcing a sparsity constraint using a bounded ℓ_1 norm. The work of [45] reformulates the principle component analysis problem as a elastic-net regularized ridge regression problem, which can be solved efficiently using least angle regression. The work of [9] proposes a convex relaxation for the sparse principle component analysis problem problem based on semidefinite programming.

One difficulty of solving Problem (1) comes from the combinational nature of the cardinality constraint. A conventional way to solve this problem is to simply replace ℓ_0 norm by its convex relaxation. Recently, non-convex approximation methods such as Schatten ℓ_p norm, re-weighted ℓ_1 norm, Capped ℓ_1 function have been proposed for acquiring better accuracy [32]. However, all these approximation methods fails to directly control the sparsity of the solution. In contrast, iterative hard thresholding maintain the sparsity of the solution by iteratively setting the small elements (in magnitude) to zero in a gradient descent manner. Due to its simplicity, it has been widely used and incorporated into the truncated power method [41] and truncated Rayleigh flow method [35].

Another difficulty of solving Problem (1) is due to the non-convexity of the objective function. One popular method to overcome this difficulty is removing the quadrat-

ic term using semidefinite programming lifting technique and reformulating (1) into the following low-rank sparse optimization problem: $\min_{\mathbf{X} \neq \mathbf{0}} tr(\mathbf{AX})/tr(\mathbf{CX})$, $s.t.$ $\mathbf{X} \succeq 0$, $rank(\mathbf{X}) = 1$, $\|\mathbf{X}\|_0 \leq s^2$. We remark that the objective function is quasilinear (hence both quasiconvex and quasiconcave), and one can constrain the denominator to be a positive constant using the scale-invariant property of the problem. Recently, convex semidefinite programming method drops the rank constraint and considers ℓ_1 relaxation for the sparsity constraint [9, 8, 20, 43]. It has been shown to achieve strong guarantee under suitable assumptions. However, such a matrix lifting technique will incur expensive computation overhead.

In summary, existing methods for solving Problem (1) suffer from the following limitations. **(i)** Semidefinite programming methods [9, 8, 20] are not scalable due to its high computational complexity for the eigenvalue decomposition. **(ii)** Convex/Non-convex approximation methods [34, 36, 33] fail to directly control the low-rank and sparse property of the solution. **(iii)** Hard thresholding methods [41, 19] only obtain weak optimality guarantee and result in poor accuracy in practice [3, 40].

Recently, the work of [5] considers a new optimality criterion which is based on Coordinate-Wise Optimality (C-WO) condition for sparse optimization. It is proven that C-WO condition is stronger than the optimality criterion based on hard thresholding. The work of [40] presents a new block-k optimal condition for general discrete optimization. It is shown to be stronger than CWO condition since it includes CWO condition as a special case with $k = 1$. Inspired by these works, we propose a new decomposition method for the sparse generalized eigenvalue problem, along with using a greedy method based on CWO [5] for finding the working set.

Contributions: This paper makes the following contributions. **(i)** We propose a new decomposition algorithm for solving the sparse generalized eigenvalue problem (see Section 3). **(ii)** We discuss two strategies to find the working set for our decomposition algorithm (see Section 4). **(iii)** We propose two methods to solve the sparse quadratic fractional programming subproblem (see Section 5). **(iv)** A convergence analysis for the decomposition method is provided (see Section 6). **(v)** Our experiments have shown that our method outperforms existing solutions in term of accuracy. (see Section 7).

Notation: All vectors are column vectors and superscript T denotes transpose. $\mathbf{X}_{i,j}$ denotes the $(i^{\text{th}}, j^{\text{th}})$ element of matrix $\mathbf{X}$ and $\mathbf{x}_i$ denotes the i-th element of vector $\mathbf{x}$. $\mathbf{e}_i$ is a unit vector with a 1 in the ith entry and 0 in all other entries. $\text{diag}(\mathbf{x})$ is a diagonal matrix formed with $\mathbf{x}$ as its principal diagonal. For any partition of the index vector $[1, 2, ..., n]$ into $[B, N]$ with $B \in \mathbb{N}^k$, $N \in \mathbb{N}^{n-k}$, we define $\mathbf{U}_B \in \mathbb{R}^{n \times k}$, $\mathbf{U}_N \in \mathbb{R}^{n \times (n-k)}$ as: $(\mathbf{U}_B)_{j,i} =$

$\left\{ \begin{smallmatrix} 1, & B(i) = j; \\ 0, & \text{else.} \end{smallmatrix} \right.$, $(\mathbf{U}_N)_{j,l} = \left\{ \begin{smallmatrix} 1, & N(l) = j; \\ 0, & \text{else.} \end{smallmatrix} \right.$. Therefore, we have $\mathbf{x}_B = \mathbf{U}_B^T \mathbf{x}$ and $\mathbf{x} = \mathbf{I}\mathbf{x} = (\mathbf{U}_B \mathbf{U}_B^T + \mathbf{U}_N \mathbf{U}_N^T)\mathbf{x} = \mathbf{U}_B \mathbf{x}_B + \mathbf{U}_N \mathbf{x}_N$. Finally, C_n^k denotes the number of possible combinations choosing k items from n.

2. Generalized Eigenvalue Problems

A number of standard and modern statistical learning models can be formulated as the sparse generalized eigenvalue problem, which we present some instances below.

• **Principle Component Analysis (PCA)**. Consider a data matrix $\mathbf{Z} \in \mathbb{R}^{m \times d}$, where each row represents an independent sample. The covariance matrix $\mathbf{\Sigma}$ is computed by $\mathbf{\Sigma} = \frac{1}{m-1} \sum_{i=1}^m (\mathbf{z}_i - \boldsymbol{\mu})(\mathbf{z}_i - \boldsymbol{\mu})^T \in \mathbb{R}^{d \times d}$, where $\mathbf{z}_i$ denotes i^{th} column of $\mathbf{Z}$ and $\boldsymbol{\mu} = \sum_{i=1}^m \mathbf{z}_i \in \mathbb{R}^d$. PCA can be cast into the following optimization problem: $\min_{\mathbf{x} \neq 0} (-\mathbf{x}^T \mathbf{\Sigma} \mathbf{x})/(\mathbf{x}^T \mathbf{x})$.

• **Fisher Discriminant Analysis (FDA)**. Given observations with two distinct classes with $\boldsymbol{\mu}_{(i)}$ and $\mathbf{\Sigma}_{(i)}$ being the mean vector and covariance matrix of class i ($i = 1$ or 2), respectively. FDA seeks a projection vector such that the between-class variance is large relative to the within-class variance, leading to the following problem: $\min_{\mathbf{x} \neq 0} \frac{-\mathbf{x}^T ((\boldsymbol{\mu}_{(1)} - \boldsymbol{\mu}_{(2)})(\boldsymbol{\mu}_{(1)} - \boldsymbol{\mu}_{(2)})^T)\mathbf{x}}{\mathbf{x}^T (\mathbf{\Sigma}_{(1)} + \mathbf{\Sigma}_{(2)})\mathbf{x}}$.

• **Canonical Correlation Analysis (CCA)**. Given two classes of data $\mathbf{X} \in \mathbb{R}^{m_1 \times d}$ and $\mathbf{Y} \in \mathbb{R}^{m_2 \times d}$, the covariance matrix between samples from $\mathbf{X}$ and $\mathbf{Y}$ can be constructed as $\mathbf{\Sigma} \triangleq \left(\begin{smallmatrix} \mathbf{\Sigma}_{xx} & \mathbf{\Sigma}_{xy} \\ \mathbf{\Sigma}_{yx} & \mathbf{\Sigma}_{yy} \end{smallmatrix} \right) \in \mathbb{R}^{(m_1+m_2) \times (m_1+m_2)}$. CCA exploits the relation of the samples by solving the following problem: $\max_{\mathbf{u} \neq 0, \ \mathbf{v} \neq 0} \mathbf{u}^T \mathbf{\Sigma}_{xy} \mathbf{v}$, $s.t.$ $\mathbf{u}^T \mathbf{\Sigma}_{xx} \mathbf{u} = \mathbf{v}^T \mathbf{\Sigma}_{yy} \mathbf{v} = 1$, where $\mathbf{A} \triangleq \left(\begin{smallmatrix} \mathbf{0} & \mathbf{\Sigma}_{xy} \\ \mathbf{\Sigma}_{yx} & \mathbf{0} \end{smallmatrix} \right)$, $\mathbf{C} \triangleq \left(\begin{smallmatrix} \mathbf{\Sigma}_{xx} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{yy} \end{smallmatrix} \right) \in \mathbb{R}^{(m_1+m_2) \times (m_1+m_2)}$, and $\mathbf{x} \triangleq [\mathbf{u}^T \mathbf{v}^T]^T$. One can rewrite CCA as the following equivalent problem: $\min_{\mathbf{x}} \frac{-\mathbf{x}^T \mathbf{A} \mathbf{x}}{\mathbf{x}^T \mathbf{C} \mathbf{x}}$.

Incorporated with the sparsity constraint, the applications listed above become special cases of the general optimization models in (1).

3. The Proposed Decomposition Algorithm

This section presents our decomposition algorithm for solving (1), which is based on the following notation of block-k optimality [40] for general non-convex constrained optimization.

Definition 1. *(Block-k Optimal Solution and Block-k Optimality Measure) (i) We denote $B \in \mathbb{N}^k$ as a vector containing k unique integers selected from $\{1, 2, ..., n\}$. We define $N \triangleq \{1, 2, ..., n\} \setminus B$, $\mathbf{x} = \mathbf{U}_B \mathbf{x}_B + \mathbf{U}_N \mathbf{x}_N$ and let*

$$\mathcal{P}(B, \mathbf{x}) \triangleq \arg\min_{\mathbf{x}_B} f(\mathbf{U}_B \mathbf{x}_B + \mathbf{U}_N \mathbf{x}_N),$$
$$s.t. \ (\mathbf{U}_B \mathbf{x}_B + \mathbf{U}_N \mathbf{x}_N) \in \Omega. \tag{2}$$

A solution $\bar{\mathbf{x}}$ is the block-k optimal solution if and only if $\bar{\mathbf{x}}_B = \mathcal{P}(B, \bar{\mathbf{x}})$ for all $|B| = k$. In other words, a solution

*is the block-k optimal solution if and only if every block co-ordinate of size k achieves the global optimal solution. **(ii)** We define $\mathcal{M}(\mathbf{x}) \triangleq \frac{1}{C_n^k} \sum_{i=1}^{C_n^k} \|\mathcal{P}(\mathcal{B}_{(i)}, \mathbf{x}) - \mathbf{x}_{\mathcal{B}_{(i)}}\|_2^2$ with $\{\mathcal{B}_{(i)}\}_{i=1}^{C_n^k}$ being all the possible combinations of the index vectors choosing k items from n with $\mathcal{B}_{(i)} \in \mathbb{N}^k$ for all i. $\mathcal{M}(\mathbf{x})$ is an optimality measure for Problem 1 in the sense that $\mathcal{M}(\bar{\mathbf{x}}) = 0$ if and only if $\bar{\mathbf{x}}$ is the block-k optimal solution.*

We describe the basic idea of the decomposition method. In each iteration t, the indices $\{1, 2, ..., n\}$ of decision variable are separated to two sets B^t and N^t, where B^t is the working set and $N^t = \{1, 2, ..., n\} \setminus B^t$. To simplify the notation, we use B instead of B^t. Therefore, we can rewrite $h(\cdot)$ and $g(\cdot)$ in Problem (1) as:

$$h(\mathbf{x}_B, \mathbf{x}_N) = \tfrac{1}{2}\mathbf{x}_B^T\mathbf{A}_{BB}\mathbf{x}_B + \tfrac{1}{2}\mathbf{x}_N^T\mathbf{A}_{NN}\mathbf{x}_N + \langle\mathbf{x}_B, \mathbf{A}_{BN}\mathbf{x}_N\rangle,$$
$$g(\mathbf{x}_B, \mathbf{x}_N) = \tfrac{1}{2}\mathbf{x}_B^T\mathbf{C}_{BB}\mathbf{x}_B + \tfrac{1}{2}\mathbf{x}_N^T\mathbf{C}_{NN}\mathbf{x}_N + \langle\mathbf{x}_B, \mathbf{C}_{BN}\mathbf{x}_N\rangle.$$

The vector $\mathbf{x}_N$ is fixed so the objective value becomes a subproblem with the variable $\mathbf{x}_B$. Our proposed algorithm iteratively solves the small-sized optimization problem with respect to the variable $\mathbf{x}_B$ as in (3) until convergence. We summarize our method in Algorithm 1.

Algorithm 1 A Decomposition Algorithm for Sparse Generalized Eigenvalue Problem as in (1).

1: Specify the working set parameter k and the proximal term parameter θ. Find an initial feasible solution $\mathbf{x}^0$ and set $t = 0$.
2: **while** not converge **do**
3:    **(S1)** Use some strategy to find a working set B whose size is k. Define $N \triangleq \{1, 2, ..., n\} \setminus B$.
4:    **(S2)** Solve the following subproblem with the variable $\mathbf{x}_B$ using combinatorial search:

$$\mathbf{x}_B^{t+1} \Leftarrow \arg\min_{\mathbf{x}_B} \frac{h(\mathbf{x}_B, \mathbf{x}_N^t) + \frac{\theta}{2}\|\mathbf{x}_B - \mathbf{x}_B^t\|_2^2}{g(\mathbf{x}_B, \mathbf{x}_N^t)}$$
$$s.t. \ \|\mathbf{x}_B\|_0 + \|\mathbf{x}_N^t\|_0 \leq s \qquad (3)$$

5:    **(S3)** Increment t by 1.
6: **end while**

Remarks. (i) The concept of block-k optimality has been introduced in [40]. This paper extends their method for minimizing convex functions to handle general non-convex objective functions. **(ii)** Algorithm 1 relies on solving a small-sized quadratic fractional problem as in (3). However, using the specific structure of the objective function and the sparsity constraint, we can develop an efficient and practical algorithm to solve it globally. **(iii)** We propose a new proximal strategy when solving the subproblem as in (3). Note that the proximal strategy is only applied to the numerator instead of to the whole objective function. This is

to guarantee sufficient descent condition and global convergence of Algorithm 1 (see Lemma 2 and Theorem 2). **(iv)** When the dimension n is small [1] and the parameter setting $\theta = 0$, $k = n$ is used, the subproblem in (3) is equivalent to Problem (1).

4. Finding the Working Set

This section shows how to find the working set (refer to Step **S1** in Algorithm 1). This problem is challenging for two aspects. **(i)** Unlike convex methods that one can find the working set using the first-order optimal condition or KKT primal-dual residual [17, 7], there is no general criteria to find the working set for non-convex problems. **(ii)** There are C_n^k possible combinations of choice for the working set of size k. One cannot expect to use the cyclic strategy and alternatingly minimize over all the possible combinations (i.e., $\{\mathcal{B}_{(i)}\}_{i=1}^{C_n^k}$) due to its high computational complexity when k is large. We propose the following two strategies to find the working set:

• **Random Strategy.** We uniformly select one combination (which contains k coordinates) from $\{\mathcal{B}_{(i)}\}_{i=1}^{C_n^k}$. In expectation, our algorithm is still guaranteed to find the block-k stationary point.

• **Swapping Strategy.** We denote $\mathcal{S}(\mathbf{x})$ and $\mathcal{Z}(\mathbf{x})$ as the index of non-zero elements and zero elements of $\mathbf{x}$, respectively. Based on the current solution $\mathbf{x}^t$, our method enumerates all the possible pairs (i, j) with $i \in \mathcal{S}(\mathbf{x}^t)$, $j \in \mathcal{Z}(\mathbf{x}^t)$ that lead to the greatest descent $\mathbf{D}_{i,j}$ by changing the two coordinates from zero/non-zero to non-zero/zero, as follows:

$$\mathbf{D}_{i,j} = \min_\beta \ f(\mathbf{x}^t + \beta\mathbf{e}_i - \mathbf{x}_j^t\mathbf{e}_j) - f(\mathbf{x}^t). \qquad (4)$$

We then pick the top pairs of coordinates that lead to the greatest descent by measuring $\mathbf{D} \in \mathbb{R}^{|\mathcal{S}(\mathbf{x})| \times |\mathcal{Z}(\mathbf{x})|}$. Specifically, we sort the elements in $\mathbf{D}$ with $\mathbf{D}_{P_1,S_1} \leq \mathbf{D}_{P_2,S_2} \leq \mathbf{D}_{P_3,S_3} \leq, ..., \mathbf{D}_{P_n,S_n}$, where $P \in \mathbb{N}^n$ and $S \in \mathbb{N}^n$ are the index vectors. Assuming that k is an even number, we simply pick the top-$(k/2)$ nonoverlapping elements of the sequence P and S respectively as the working set.

We now discuss how to solve (4) to obtain $\mathbf{D}_{i,j}$. We start from the following lemma.

Lemma 1. *We consider the following one-dimensional optimization problem:*

$$\beta^* = \arg\min_\beta \psi(\beta) \triangleq \frac{\frac{1}{2}\bar{a}\beta^2 + \bar{b}\beta + \bar{c}}{\frac{1}{2}\bar{r}\beta^2 + \bar{s}\beta + \bar{t}}, \ s.t. \ \beta \geq \bar{L} \qquad (5)$$

*Assume that $\forall \beta \geq \bar{L}$, $\tau \triangleq \frac{1}{2}\bar{r}\beta^2 + \bar{s}\beta + \bar{t} > 0$ and the optimal solution is bounded. We have: **(i)** Problem (5) admits a closed-form solution as: $\beta^* = \arg\min_\beta \ f(\beta)$, $\beta \in$*

$\{\Pi(\beta_1), \Pi(\beta_2)\}$, *where* $\beta_1 = (-\vartheta - \sqrt{\vartheta^2 - 2\pi\iota})/\pi$, $\beta_2 = (-\vartheta + \sqrt{\vartheta^2 - 2\pi\iota})/\pi$, $\pi \triangleq \bar{a}\bar{s} - \bar{b}\bar{r}$, $\vartheta \triangleq \bar{a}\bar{t} - \bar{c}\bar{r}$, $\iota \triangleq \bar{t}\bar{b} + \bar{c}\bar{s}$, *and* $\Pi(\beta) \triangleq \max(\bar{L}, \beta)$. *(ii) Problem (5) contains one unique optimal solution.*

Proof. **(i)** Dropping the bound constraint and setting the gradient of $\psi(\beta)$ to zero, we have $0 = \psi'(\beta) = ((\bar{a}\beta + \bar{b})(\frac{1}{2}\bar{r}\beta^2 + \bar{s}\beta + \bar{t}) - (\frac{1}{2}\bar{a}\beta^2 + \bar{b}\beta + \bar{c})(\bar{r}\beta + \bar{s}))/\tau^2$. Noticing $\tau > 0$, we obtain the following first-order optimal condition for ψ: $0 = (\bar{a}\beta + \bar{b})(\frac{1}{2}\bar{r}\beta^2 + \bar{s}\beta + \bar{t}) - (\frac{1}{2}\bar{a}\beta^2 + \bar{b}\beta + \bar{c})(\bar{r}\beta + \bar{s})$. It can be simplified as: $0 = \frac{1}{2}\beta^2\pi + \beta\vartheta + \iota$. Solving this equation, we have two solutions β_1 and β_2. We select the one between $\Pi(\beta_1)$ and $\Pi(\beta_2)$ that leads to a lower objective value as the optimal solution. **(ii)** It can be proven by contradiction. We omit the one-sided bound constraint since it does not effect the uniqueness of the optimal solution. Assume that there exist two optimal solutions x and y to (1) that lead to the same objective value ϑ. According to the first-order and second-order optimal condition [10, 42], we have: $(\bar{a} - \vartheta\bar{r})x = -\bar{b} + \vartheta\bar{s}$, $(\bar{a} - \vartheta\bar{r})y = -\bar{b} + \vartheta\bar{s}$, $(\bar{a} - \vartheta\bar{r}) > 0$, which leads to the following contradiction: $\frac{\vartheta\bar{s} - \bar{b}}{\bar{a} - \vartheta\bar{r}} = x \neq y = \frac{\vartheta\bar{s} - \bar{b}}{\bar{a} - \vartheta\bar{r}}$. Therefore, (11) contains one unique optimal solution. Please refer to Figure 1. □
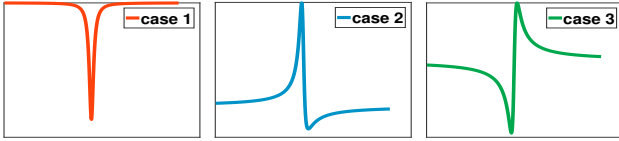

Figure 1: Geometric interpretation for the one-dimensional quadratic fractional problem. Using the l'Hopital's Rule, we have $\lim_{\beta \to +\infty} \psi(\beta) = \lim_{\beta \to -\infty} \psi(\beta) = \frac{\bar{a}}{\bar{r}}$. Since the optimal solution is bounded and the problem at most contains two critical points, we only have the three cases above. Clearly, there exists one unique optimal solution.

Letting $\mathbf{v} \triangleq \mathbf{x}^t - \mathbf{x}_j^t\mathbf{e}_j$, we obtain: $\min_\beta f(\mathbf{v} + \beta\mathbf{e}_i) = \min_\beta \frac{\frac{1}{2}(\mathbf{v}+\beta\mathbf{e}_i)^T\mathbf{A}(\mathbf{v}+\beta\mathbf{e}_i)}{\frac{1}{2}(\mathbf{v}+\alpha\mathbf{e}_i)^T\mathbf{C}(\mathbf{v}+\beta\mathbf{e}_i)}$. By applying Lemma 1 with $\bar{L} = -\infty$, $\bar{a} = \mathbf{A}_{i,i}$, $\bar{b} = (\mathbf{Av})_i$, $\bar{c} = \frac{1}{2}\mathbf{v}^T\mathbf{Av}$, $\bar{r} = \mathbf{C}_{i,i}$, $\bar{s} = (\mathbf{Cv})_i$, $\bar{t} = \frac{1}{2}\mathbf{v}^T\mathbf{Cv}$, we obtain the global optimal solution for (4).

5. Solving the Subproblem

The subproblem (3) in Algorithm 1 (refer to Step **S2** in Algorithm 1) reduces to the following quadratic fractional programming problem:

$$\mathbf{z}^* = \arg\min_{\|\mathbf{z}\|_0 \leq q} \ p(\mathbf{z}) \triangleq \frac{\frac{1}{2}\mathbf{z}^T\bar{\mathbf{Q}}\mathbf{z} + \bar{\mathbf{p}}^T\mathbf{z} + \bar{w}}{\frac{1}{2}\mathbf{z}^T\bar{\mathbf{R}}\mathbf{z} + \bar{\mathbf{c}}^T\mathbf{z} + \bar{v}}, \quad (6)$$

where $\mathbf{z} \in \mathbb{R}^k$, $\bar{\mathbf{Q}} = \mathbf{A}_{BB} + \theta\mathbf{I}$, $\bar{\mathbf{p}} = \mathbf{A}_{BN}\mathbf{x}_N - \theta\mathbf{x}_B^t$, $\bar{w} = \frac{1}{2}\mathbf{x}_N^T\mathbf{A}_{NN}\mathbf{x}_N + \frac{\theta}{2}\|\mathbf{x}_B^t\|_2^2$, $\bar{\mathbf{R}} = \mathbf{C}_{BB}$, $\bar{\mathbf{c}} = \mathbf{C}_{BN}\mathbf{x}_N$, , $\bar{v} = \frac{1}{2}\mathbf{x}_N^T\mathbf{C}_{NN}\mathbf{x}_N$, $q = s - \|\mathbf{x}_N\|_0$.

Problem (6) is equally NP-hard due to the combinatorial constraint $\|\mathbf{z}\|_0 \leq q$. Inspired by the work of [40], we develop an exhaustive tree/combinatorial search algorithm to solve it. Specifically, we consider to solve the following optimization problem: $\min_{\mathbf{z} \in \mathbb{R}^k} \ p(\mathbf{z})$, $s.t. \ \mathbf{z}_K = 0$, where K has $\sum_{i=0}^q C_k^i$ possible choices for the coordinates. We systematically enumerate the full binary tree for K to obtain all possible candidate solutions for $\mathbf{z}$ and then pick the best one that leads to the lowest objective value as the optimal solution. In other words, we need to solve the following quadratic fractional programming problem with $m \triangleq k - |K|$ variables:

$$\mathbf{y}^* = \arg\min_{\mathbf{y}} \ \mathcal{L}(\mathbf{y}) \triangleq \frac{u(\mathbf{y})}{q(\mathbf{y})} \triangleq \frac{\frac{1}{2}\mathbf{y}^T\mathbf{Qy} + \mathbf{p}^T\mathbf{y} + w}{\frac{1}{2}\mathbf{y}^T\mathbf{Ry} + \mathbf{c}^T\mathbf{y} + v}, \quad (7)$$

where $\mathbf{y} \in \mathbb{R}^m$. The optimal solution of (6) can be computed as $\mathbf{z}_K^* = \mathbf{0}$, $\mathbf{z}_{\bar{K}}^* = \mathbf{y}^*$ with $\bar{K} = \{1, 2, ..., k\} \setminus K$. Therefore, if we find the global optimal solution of (7), we find the global optimal solution of (6) as well.

The non-convex problem in (7) is still challenging. For solving it, we present two methods, namely a bisection search method and a coordinate descent method, which are of independent research interest.

5.1. A Bisection Search Method

This subsection presents a bisection search method for finding the global optimal solution of Problem (7).

We now discuss the relationship between this fractional programming problem and the following parametric programming problem [10]:

$$\mathcal{J}(\alpha) = 0, \text{ with } \mathcal{J}(\alpha) \triangleq \min_{\mathbf{y}} \ u(\mathbf{y}) - \alpha q(\mathbf{y})$$

This is a feasibility problem with respect to α: $\mathcal{J}(\alpha) = u(y^*(\alpha)) - \alpha q(y^*(\alpha)) = 0$, where $y^*(\alpha) \in \mathbb{R}^m$ is defined as $y^*(\alpha) \triangleq \arg\min_{\mathbf{y}} \ u(\mathbf{y}) - \alpha q(\mathbf{y})$.

The following theorem sheds some theoretic lights for the original non-convex problem in (7).

Theorem 1. *We have the following results.* **(i)** *It holds that:* $\lambda_{\min}(\mathbf{Z}) \leq \min_{\mathbf{y}} \ \mathcal{L}(\mathbf{y}) < \lambda_{\min}(\mathbf{O})$, *with* $\mathbf{O} \triangleq \mathbf{R}^{-1/2}\mathbf{Q}\mathbf{R}^{-1/2}$, $\gamma \triangleq 2v - \|\mathbf{R}^{-1/2}\mathbf{c}\|_2^2 > 0$, $\mathbf{g} \triangleq \mathbf{R}^{-1/2}\mathbf{p} - \mathbf{R}^{-1/2}\mathbf{Q}\mathbf{R}^{-1}\mathbf{c}$, $\delta \triangleq \mathbf{c}^T\mathbf{R}^{-1}\mathbf{Q}\mathbf{R}^{-1}\mathbf{c} - 2\mathbf{c}^T\mathbf{R}^{-1}\mathbf{p} + 2w$, *and* $\mathbf{Z} \triangleq \begin{pmatrix} \mathbf{O} & \mathbf{g}/\sqrt{\gamma} \\ \mathbf{g}^T/\sqrt{\gamma} & \delta/\gamma \end{pmatrix}$. **(ii)** *Let* $\mathbf{O} = \mathbf{U}diag(\mathbf{d})\mathbf{U}^T$ *be the eigenvalue decomposition of* $\mathbf{O}$. *The function* $\mathcal{J}(\alpha)$ *can be rewritten as*

$$\mathcal{J}(\alpha) = \frac{1}{2}\delta - \frac{1}{2}\alpha\gamma - \frac{1}{2}\sum_i^m \frac{\mathbf{a}_i^2}{\mathbf{d}_i - \alpha}, \text{ with } \mathbf{a} = \mathbf{U}^T\mathbf{g} \quad (8)$$

and it is monotonically decreasing on the range $\lambda_{\min}(\mathbf{Z}) \leq \alpha < \lambda_{\min}(\mathbf{O})$. *The optimal solution can be computed as* $\mathbf{y}^* = \mathbf{R}^{-1/2}(\mathbf{u}^* - \mathbf{R}^{-1/2}\mathbf{c})$, *with* $\mathbf{u}^* = -(\mathbf{O} - \alpha^*\mathbf{I})^{-1}\mathbf{g}$ *and* α^* *being the unique root of the equation* $\mathcal{J}(\alpha) = 0$ *on the range* $\lambda_{\min}(\mathbf{Z}) \leq \alpha < \lambda_{\min}(\mathbf{O})$.

Proof. **(i)** Firstly, it is not hard to notice that Program (7) is equivalent to the following problem:

$$\min_{\mathbf{y}} \mathcal{L}(\mathbf{y}) = \min_{\mathbf{d}} \frac{\frac{1}{2}(\mathbf{R}^{-1/2}\mathbf{d})^T\mathbf{Q}(\mathbf{R}^{-1/2}\mathbf{d})+\mathbf{p}^T(\mathbf{R}^{-1/2}\mathbf{d})+w}{\frac{1}{2}(\mathbf{R}^{-1/2}\mathbf{d})^T\mathbf{R}(\mathbf{R}^{-1/2}\mathbf{d})+\mathbf{c}^T(\mathbf{R}^{-1/2}\mathbf{d})+v}$$
$$= \min_{\mathbf{d}} \frac{\frac{1}{2}\mathbf{d}^T\mathbf{O}\mathbf{d}+\mathbf{d}^T(\mathbf{R}^{-1/2}\mathbf{p})+w}{\frac{1}{2}\|\mathbf{d}\|_2^2+\mathbf{d}^T(\mathbf{R}^{-1/2}\mathbf{c})+v}$$
$$= \min_{\mathbf{d}} \frac{\frac{1}{2}\mathbf{d}^T\mathbf{O}\mathbf{d}+\mathbf{d}^T(\mathbf{R}^{-1/2}\mathbf{p})+w}{\frac{1}{2}\|\mathbf{d}+\mathbf{R}^{-1/2}\mathbf{c}\|_2^2+v-\frac{1}{2}\|\mathbf{R}^{-1/2}\mathbf{c}\|_2^2}$$
$$= \min_{\mathbf{u}} \frac{\frac{1}{2}\mathbf{u}^T\mathbf{O}\mathbf{u}+\mathbf{u}^T\mathbf{g}+\frac{1}{2}\delta}{\frac{1}{2}\|\mathbf{u}\|_2^2+\frac{1}{2}\gamma}, \tag{9}$$

where the first step uses the variable substitution that $\mathbf{y} = \mathbf{R}^{-1/2}\mathbf{d}$; the third step uses the transformation that $\mathbf{u} = \mathbf{d}+\mathbf{R}^{-1/2}\mathbf{c}$. We notice that the denominator is always strictly positive for all decision variables. Letting $\mathbf{u} = \mathbf{0}$ in (9), we obtain $\frac{1}{2}\gamma > 0$. We naturally obtain the upper bound for $\min_{\mathbf{y}} \mathcal{L}(\mathbf{y})$:

$$\min_{\mathbf{y}} \mathcal{L}(\mathbf{y}) = \min_{\mathbf{u},\eta=\sqrt{\gamma}} \frac{\frac{1}{2}\mathbf{u}^T\mathbf{O}\mathbf{u}+\frac{1}{\sqrt{\gamma}}\mathbf{u}^T\mathbf{g}\eta+\frac{\delta}{2\gamma}\eta^2}{\frac{1}{2}\|\mathbf{u}\|_2^2+\frac{1}{2}\eta^2}$$
$$\geq \min_{\mathbf{u},\eta} \frac{\frac{1}{2}\mathbf{u}^T\mathbf{O}\mathbf{u}+\frac{1}{\sqrt{\gamma}}\mathbf{u}^T\mathbf{g}\eta+\frac{\delta}{2\gamma}\eta^2}{\frac{1}{2}\|\mathbf{u}\|_2^2+\frac{1}{2}\eta^2}$$
$$= \min_{\mathbf{u},\eta} \frac{\frac{1}{2}[\mathbf{u}^T \mid \eta^T]^T\mathbf{Z}[\mathbf{u}^T \mid \eta^T]}{\frac{1}{2}\|\mathbf{u}\|_2^2+\frac{1}{2}\eta^2} = \lambda_{\min}(\mathbf{Z}),$$

where the first inequality uses the fact that $\min_{\mathbf{x}} f(\mathbf{x}) \leq \min_{\mathbf{x}\in\Psi} f(\mathbf{x})$ for all $f(\cdot)$ and Ψ.

We now derive the upper bound of $\min_{\mathbf{y}} \mathcal{L}(\mathbf{y})$. Since the objective function $\mathcal{J}(\alpha)$ is always bounded, there must exist α with $\mathbf{Q}-\alpha\mathbf{R} \succ 0$, such that the value of $\mathbf{y}$ minimizing the function $(u(\mathbf{y}) - \alpha q(\mathbf{y}))$. Therefore, we have $\mathbf{Q} - \alpha\mathbf{R} \succ 0 \Rightarrow \mathbf{Q} - \alpha\mathbf{R}^{1/2}\mathbf{I}\mathbf{R}^{1/2} \succ 0 \Rightarrow \mathbf{R}^{-1/2}\mathbf{Q}\mathbf{R}^{-1/2} - \alpha\mathbf{I} \succ 0 \Rightarrow \alpha < \lambda_{\min}(\mathbf{O})$.

(ii) Using the result of (9), we can rewrite $\mathcal{J}(\alpha)$ as:

$$\mathcal{J}(\alpha) = \min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T\mathbf{O}\mathbf{u} + \mathbf{u}^T\mathbf{g} + \frac{1}{2}\delta - \alpha\left(\frac{1}{2}\|\mathbf{u}\|_2^2 + \frac{1}{2}\gamma\right)$$
$$= \min_{\mathbf{u}} \frac{1}{2}\mathbf{u}^T(\mathbf{O} - \alpha I)\mathbf{u} + \mathbf{u}^T\mathbf{g} + \frac{1}{2}\delta - \frac{\alpha\gamma}{2}.$$

Solving the quadratic optimization with respect to $\mathbf{u}$ we have $\mathbf{u}^* = -(\mathbf{O} - \alpha\mathbf{I})\mathbf{g}$. Thus, we can repress $\mathcal{J}(\alpha)$ as: $\mathcal{J}(\alpha) = -\frac{1}{2}\mathbf{g}^T(\mathbf{O} - \alpha I)^{-1}\mathbf{g} + \frac{1}{2}\delta - \frac{\alpha\gamma}{2}$. Since it holds that $\mathbf{g}^T(\mathbf{O} - \alpha I)^{-1}\mathbf{g} = \mathbf{g}^T\mathbf{U}^T\text{diag}(1 \div (\mathbf{d} - \alpha))\mathbf{U}\mathbf{g}$ with $\div$ denoting the element-wise division between two vectors, we obtain (8). Noticing that the first-order and second-order gradient of $\mathcal{J}(\alpha)$ with respect to α can be computed as: $\mathcal{J}'(\alpha) = -\frac{1}{2}\sum_i^m (\frac{\mathbf{a}_i}{\mathbf{d}_i-\alpha})^2 - \frac{\gamma}{2}$, $\mathcal{J}''(\alpha) = -\sum_i^m(\mathbf{a}_i^2/(\mathbf{d}_i - \alpha)^3)$ and $\gamma > 0$, we obtain $\mathcal{J}'(\alpha) < 0$ and $\mathcal{J}''(\alpha) \leq 0$. Thus, the function $\mathcal{J}(\alpha)$ is concave and monotonically decreasing on the range $\lambda_{\min}(\mathbf{Z}) \leq \alpha < \lambda_{\min}(\mathbf{O})$, and there exists a unique root of the equation $\mathcal{J}(\alpha) = 0$ on the range $\lambda_{\min}(\mathbf{Z}) \leq \alpha < \lambda_{\min}(\mathbf{O})$.

$\square$

Based on Theorem 1, we now present a bisection method for solving Problem (7). For notation convenience, we define $\underline{\alpha} \triangleq \lambda_{\min}(\mathbf{Z})$ and $\overline{\alpha} \triangleq \lambda_{\max}(\mathbf{O}) - \epsilon$, where ϵ de-

notes the machine precision parameter which is sufficiently small. Due to the monotonically decreasing property of $\mathcal{J}(\alpha)$, we can solve (8) by checking where the sign of the left-hand side changes. Specifically, we consider the following three cases for $\mathcal{J}(\alpha)$ on the range $\underline{\alpha} \leq \alpha \leq \overline{\alpha}$: **(a)** $\mathcal{J}(\underline{\alpha}) \geq \mathcal{J}(\overline{\alpha}) \geq 0$, **(b)** $0 \geq \mathcal{J}(\underline{\alpha}) \geq \mathcal{J}(\overline{\alpha})$, and **(c)** $\mathcal{J}(\underline{\alpha}) \geq 0 \geq \mathcal{J}(\overline{\alpha})$. For case **(a)** and **(b)**, we can directly return $\overline{\alpha}$ and $\underline{\alpha}$ as the optimal solution, respectively. We now consider case **(c)**. By the Rolle mean value theorem, there always exists an $\alpha^* \in [\underline{\alpha}, \overline{\alpha}]$ such that $\mathcal{J}(\alpha^*) = 0$. Thus, we can define and initialize the lower bound $lb = \underline{\alpha}$ and the upper bound $ub = \overline{\alpha}$. We then perform the following loop until the optimal solution $\alpha^* = mid$ with $\mathcal{J}(mid) \approx 0$ is found: $\{mid = (lb + ub)/2$, if$(\mathcal{J}(mid) > 0)$ $lb = mid$; else $ub = mid;\}$. Such a bisection scheme is guaranteed to find the optimal solution within $\mathcal{O}(\log_2((\overline{\alpha}-\underline{\alpha})/\varepsilon))$ iterations that $ub \leq lb + \varepsilon$ [6].

Remarks. **(i)** To our knowledge, this is the first algorithm for unconstrained quadratic fractional programming with global optimal guarantee. The work of [12] also discusses a bisection search method for the ratio of trace problem, but it can not solve our general quadratic fractional programming problem. The classical Dinkelbach's method [10, 39] can solve our problem, but it only finds a stationary solution for the non-convex problem. Our results are based on the monotone property of the associated parametric programming problem in a restricted domain. **(ii)** The unconstrained fractional quadratic program can be solved to optimality by linear semidefinite programming and it is related to the S-lemma for the quadratically constrained quadratic program [29]. In this paper, we show that it can be solved using a bisection search method. This method has the merit that it is simple and easy to implement. In addition, it is efficient and it does not require iterative eigenvalue decomposition as in the semidefinite programming lifting techniques. **(iii)** The matrix $\mathbf{O}$ is a $n \times n$ principal sub-matrix of $\mathbf{Z}$. Using Theorem 4.3.17 in [13], it always holds that $\boldsymbol{\lambda}_1(\mathbf{Z}) \leq \boldsymbol{\lambda}_1(\mathbf{O}) \leq \boldsymbol{\lambda}_2(\mathbf{Z}) \leq ... \leq \boldsymbol{\lambda}_{n-1}(\mathbf{Z}) \leq \boldsymbol{\lambda}_{n-1}(\mathbf{O}) \leq \boldsymbol{\lambda}_n(\mathbf{Z})$, where $\boldsymbol{\lambda}(\mathbf{X})$ denotes the eigenvalues of $\mathbf{X}$ in increasing order. Thus, the bound for the α^* is tight.

5.2. A Coordinate Descent Method

This subsection presents a simple coordinate descent method [37, 14, 15, 38, 22] for solving Problem (7). Although it can not guarantee to find the global optimal solution, it has many merits. **(i)** It is able to incorporate additional bound constraints. **(ii)** It is numerically robust and does not require additional eigenvalue solvers. **(iii)** It is guaranteed to converge to a coordinate-wise minimum point for our specific problem (see Proposition 1 below).

To illustrate the merits of the coordinate descent method, we consider incorporating the bound constraint $\mathbf{x} \geq \bar{L}$ on

the solution for Problem (1) [2]. Our decomposition algorithm for finding the working set and strategies for handling the NP-hard ℓ_0 norm directly follow and what one needs is to replace (7) and solve the following problem:

$$\min_{\mathbf{y} \in \mathbb{R}^m} \ \mathcal{L}(\mathbf{y}) \triangleq \frac{\frac{1}{2}\mathbf{y}^T\mathbf{Q}\mathbf{y}+\mathbf{p}^T\mathbf{y}+w}{\frac{1}{2}\mathbf{y}^T\mathbf{R}\mathbf{y}+\mathbf{c}^T\mathbf{y}+v}, \ s.t. \ \mathbf{y} \geq \hat{L} \quad (10)$$

for some constant $\hat{L}$.

The coordinate descent method iteratively picks a coordinate $i \in \{1, 2, ..., m\}$ and solves the following one dimensional subproblem based on its current solution $\mathbf{y}^j$ with $j = 0, 1, ...\infty$:

$$\beta^* = \arg\min_\beta \ \mathcal{L}(\mathbf{y}^j + \beta \mathbf{e}_i), \ s.t. \ \mathbf{y}_i^j + \beta \geq \hat{L} \quad (11)$$

where j is the iteration counter for the coordinate descent algorithm. Problem (11) reduces to the one-dimensional subproblem as in Lemma 1 with suitable parameters. In every iteration j, once the optimal solution β^* in (11) is found, the intermediate solution for (10) is updated via $\mathbf{y}_i^{j+1} \Leftarrow \mathbf{y}_i^j + \beta^*$. There are several ways and orders to decide which coordinates to update in the literature. **(i)** Cyclic order strategy runs all coordinates in cyclic order, i.e., $1 \rightarrow 2 \rightarrow ... \rightarrow m \rightarrow 1$. **(ii)** Random sampling strategy randomly selects one coordinate to update (sample with replacement). **(iii)** Gauss-Southwell strategy picks coordinate i such that $i = \arg\max_{1 \leq t \leq m} |\bar{\nabla}\mathcal{L}(\mathbf{x}^j)|_t$, with $\bar{\nabla}\mathcal{L}(\mathbf{x}) \in \mathbb{R}^m$ being the projected gradient of $\mathcal{L}$ at $\mathbf{x}$ [23]: $\bar{\nabla}\mathcal{L}(\mathbf{x})_i = \begin{cases} \nabla\mathcal{L}(\mathbf{x})_i, & \mathbf{x}_i > \hat{L}; \\ \min(0, \nabla\mathcal{L}(\mathbf{x})_i), & \mathbf{x}_i = \hat{L}; \end{cases}$ and $\nabla\mathcal{L}(\mathbf{x})$ being the gradient of $\nabla\mathcal{L}$ at $\mathbf{x}$. Note that $\bar{\nabla}\mathcal{L}(\dot{\mathbf{x}}) = \mathbf{0}$ implies $\dot{\mathbf{x}}$ is a first-order stationary point.

We now present our convergence result of the coordinate descent method for solving (10), which is an extension of Theorem 4.1 in [37]. Some proofs can be found in the **Appendix**.

Proposition 1. *When the cyclic order strategy is used, coordinate descent method is guaranteed to converge to a coordinate-wise minimum of Problem (10) that* $\forall i$, $\mathbf{y}_i^* = \arg\min_{\alpha \geq \hat{L}} \ \mathcal{L}(\mathbf{y}_i^* + \alpha \mathbf{e}_i)$.

Remarks. **(i)** Convergence of the coordinate descent method requires a unique solution in each minimization step; otherwise, it may cycle indefinitely. A simple but intriguing example is given in [30]. One good feature of the non-convex problem in (10) is that its associated one-dimensional subproblem in (11) only contains one unique optimal solution (see part (ii) in Lemma 1). This is different from the work of [38] where their one-dimensional subproblem may have multiple optimal solutions and cause divergence. **(ii)** Coordinate descent method is guaranteed to produce a coordinate-wise stationary point which is stronger than the full gradient projection method. Note

---

[2]This is useful in sparse non-negative PCA [2].

that any coordinate-wise stationary point $\mathbf{x}^*$ that $\forall i$, $\mathbf{x}_i^* = \arg\min_{\alpha \geq \hat{L}} \ \mathcal{L}(\mathbf{x}_i^* + \alpha \mathbf{e}_i)$ also satisfies the first-order optimal condition with $\bar{\nabla}\mathcal{L}(\mathbf{x}^*) = 0$. However, the reverse is not true. This implies that the coordinate descent method can exploit possible higher order derivatives to escape saddle points for the non-convex problem.

6. Convergence Analysis of Algorithm 1

This section presents the convergence analysis of Algorithm 1. We assume that $\{f(\mathbf{x}^t)\}_{t=0}^\infty$ is generated by Algorithm 1 and the solution is bounded with $0 < \|\mathbf{x}^t\| < \infty$ for all t throughout this section. We first present the following lemma.

Lemma 2. *(**Sufficient Decrease Condition**) It holds that:* $f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \leq \frac{-\theta\|\mathbf{x}^{t+1}-\mathbf{x}^t\|_2^2}{(\mathbf{x}^{t+1})^T\mathbf{C}\mathbf{x}^{t+1}}$.

Remarks. The proximal term in the numerator in (3) is necessary for our non-convex problem since it guarantees sufficient decrease condition which is important for convergence.

Now we present our main convergence result.

Theorem 2. ***Convergence Properties of Algorithm 1.*** *Assume that the subproblem in (3) is solved globally, and there exists a constant σ such that $\mathbf{x}^t\mathbf{C}\mathbf{x}^t \geq \sigma > 0$ for all t. We have the following results.*

(i) When the random strategy is used to find the working set, we have $\lim_{t\to\infty} \mathbb{E}[\|\mathbf{x}^{t+1} - \mathbf{x}^t\|] = 0$ and Algorithm 1 converges to the block-k stationary point in expectation.

(ii) When the swapping strategy is used to find the working set with $k \geq 2$, we have $\lim_{t\to\infty} \|\mathbf{x}^{t+1} - \mathbf{x}^t\| = 0$ and Algorithm 1 converges to the block-2 stationary point deterministically.

Remarks. **(i)** Thanks to the fact that the denominator is positive and the objective function is quadratic fractional, our algorithm is still guaranteed to convergence even in the presence of non-convexity. **(ii)** We propose using a swapping strategy to find the working set which enumerates all possible swaps for all pairs of coordinates to find the greatest descent. One good feature of this strategy is that it achieves optimality guarantee which is no worse than Beck and Vaisbourd's coordinate-wise optimality condition [5].

7. Experiments

This section demonstrates the efficacy of the proposed decomposition algorithm by considering three important applications (i.e., sparse PCA, sparse FDA, and sparse CCA) on synthetic and real-world data sets.

● **Data Sets**. **(i)** We consider four real-world data sets: 'a1a', 'w1a', 'w2a', and 'madelon'. We randomly select a
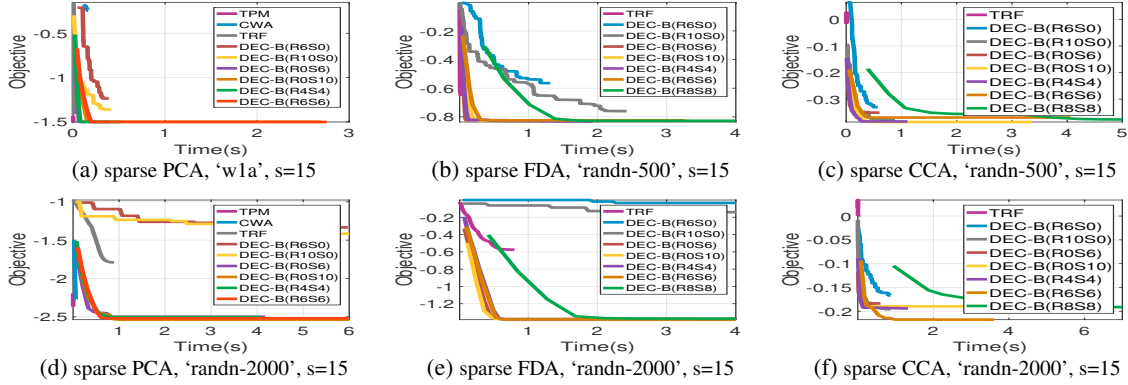
Figure 1 Convergence behavior of different methods for sparse PCA (left column), sparse FDA (middle column), and sparse CCA (right column).

subset of examples from the original data sets [3]. The size of the data sets used in our experiments are 2000×119, 2000×300, 2000×300, 2000×112, respectively. **(ii)** We also use a similar method as in [5] to generate synthetic Gaussian random data sets. Specifically, we produce the feature matrix $\mathbf{X} \in \mathbb{R}^{m \times d}$ and the label vector $\mathbf{y} \in \mathbb{R}^m$ as follows: $\mathbf{X} = \text{randn}(m, d)$, $\mathbf{y} = \text{sign}(\text{randn}(m, 1))$, where $\text{randn}(m, d)$ is a function that returns a standard Gaussian random matrix of size $m \times d$ and sign is a signum function. We fix $m = 300$ and consider different values for $d = \{100, 500, 1500, 2000\}$. We denote the data sets as 'randn-d' and place the results in the **Appendix**.

Based on $\mathbf{X}$ and $\mathbf{y}$, we generate the matrices $\mathbf{A}$ and $\mathbf{C}$ in Problem (1) for different applications (see Section 2). Note that the resulting size of the sparse generalized eigenvalue problem for sparse PCA, sparse FDA, and sparse CCA are d, d, and m, respectively. We vary the sparsity parameter $s \in \{4, 8, 12, ..., 40\}$ and report the objective values for Problem (1).

• **Compared Methods**. We compare the following methods. **(i)** Truncated Power Method (TPM) [41] [4] iteratively and greedily decreases the objective while maintaining the desired sparsity for the solutions by hard thresholding truncation. **(ii)** Coordinate-Wise Algorithm (CWA) [3, 5] [5] iteratively performs an optimization step with respect to two coordinates, where the coordinates that need to be altered are chosen to be the ones that produce the maximal decrease among all possible alternatives. **(iii)** Truncated Rayleigh Flow (TRF) [35] iteratively updates the solution using the gradient of the generalized Rayleigh quotient and performs a truncation operation to achieve sparsity. **(iv)** Quadratic Majorization Method (QMM) [32] [6] approximates the ℓ_0-norm by a continuous surrogate function and iteratively majorizes the surrogate function by a

quadratic separable function, which at each iteration reduces to a regular generalized eigenvalue problem. Using different smooth non-convex approximation functions, they develop different versions of QMM (QMM-exp, QMM-log, QMM-ℓ_p, QMM-ℓ_0). Since their methods only solve a regularized problem and fail to control the sparsity of the solution, we use a simple bisection search to find the best regulation parameter and report the lowest objective value after hard thresholding. **(v)** The proposed decomposition method (denoted as DEC) is included for comparisons. We use DEC-B(Ri-Sj) and DEC-C(Ri-Sj) to denote our method based on a **B**isection search method and a **C**oordinate descent method, respectively, along with selecting i coordinate using the **R**andom strategy and j coordinates using the **S**wapping strategy. In each iteration, we compute $r_t = (f(\mathbf{x}^t) - f(\mathbf{x}^{t+1})) / f(\mathbf{x}^t)$. We let Algorithm 1 run up to T iterations and stop it at iteration $t < T$ if $\text{mean}([r_{t-\min(t,M)+1}, r_{t-min(t,M)+2}, ..., r_t]) \leq \epsilon$. The default parameter $(\theta, \epsilon, M, T) = (10^{-5}, 10^{-5}, 50, 1000)$ is used. All codes are implemented in MATLAB on an Intel 3.20GHz CPU with 8 GB RAM [7]. Only DEC-C is developed in C and wrapped into our MATLAB code, since it uses an elementwise loop which is inefficient in native MATLAB.

We remark that both (a) and (b) are only designed for sparse PCA with $\mathbf{C} = \mathbf{I}$. We do not compare against the DC programming algorithms [34, 36] since they fail to control the sparsity of the solution and result in worse accuracy than QMM (see [32]).

• **Convergence Behavior**. We show the convergence behavior for different methods in Figure 1. We do not include the results of QMM since it fails to control the sparsity of the solution. Due to space limitation, we only report the results of DEC-B in this set of experiments. We have the following observations. **(i)** The methods {TPM, CWA, TRF} converge within one second and they are faster than

---

[3] https://www.csie.ntu.edu.tw/~cjlin/libsvm/
[4] code: sites.google.com/site/xtyuan1980/
[5] code: sites.google.com/site/amirbeck314/
[6] code: https://junxiaosong.github.io/

[7] For the purpose of reproducibility, we provide our code in the authors' research webpage.
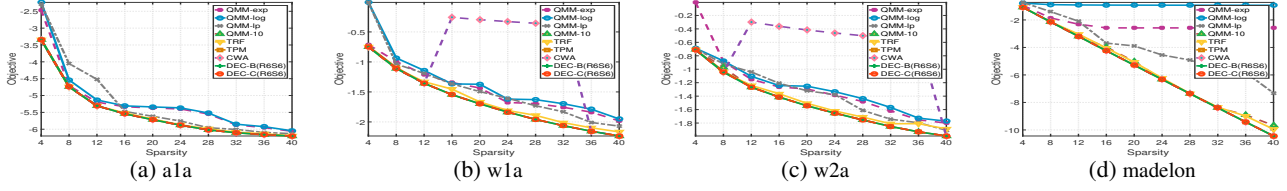
Figure 2: Accuracy of different methods on different data sets for sparse PCA problem with varying the cardinalities.
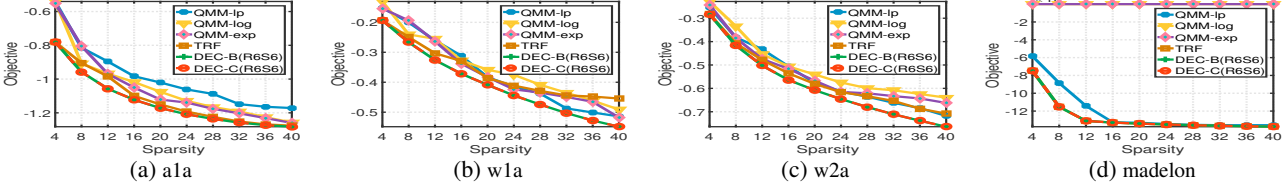

Figure 3: Accuracy of different methods on different data sets for sparse FDA problem with varying cardinalities.
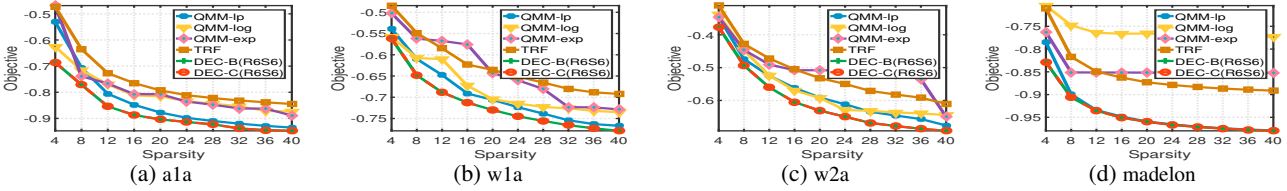

Figure 4: Accuracy of different methods on different data sets for sparse CCA problem with varying cardinalities.
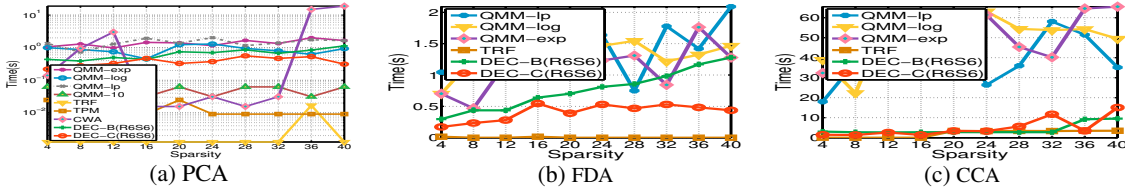

Figure 5 Comparison of the computing time for different methods on 'w1a' data set with varying cardinalities.

DEC. However, they get stuck into poor local minima and result in much worse accuracy than DEC. **(ii)** The objective values of DEC stabilize after less than 5 seconds, which means it has converged, and the decrease of the objective value is negligible afterwards. This implies that one may use a looser stopping criterion without sacrificing accuracy. **(iii)** DEC-B(R6S0) and DEC-B(R8S8) converge slowly, and it seems that DEC-B(R6S6) finds a good trade-off between efficiency and effectiveness. **(iv)** DEC-B(R10S0) achieves a lower objective value than DEC-B(R6S0). This is reasonable since a larger k in the block-k optimality condition implies a stronger stationary point. **(v)** {DEC-B(R6S0), DEC-B(R10S0)} achieve larger objective values than {DEC-B(R0S6), DEC-B(R0S10)}, which implies that the swapping strategy plays an indispensable role in DEC.

● **Experimental Results**. We show the experimental results for sparse PCA, sparse FDA, and sparse CCA in Figure 2, 3 and 4, respectively. Several conclusions can be drawn. **(i)** CWA generally outperforms {TPM, TRF, QMM}. **(i-**

i) CWA is not stable and generates much worse results on 'w1a' and 'w2a'. **(iii)** The proposed method DEC still outperforms CWA and achieves the lowest objective values. **(iv)** Both DEC-B and DEC-C perform similarly. This is because coordinate descent methods find a desirable solution for the quadratic fractional programming problem.

● **Computational Efficiency**. We demonstrate a comparison of the actual computing time for different methods on 'w1a' data set in Figure 5. Two conclusions can be drawn. **(i)** DEC takes less than 15 seconds to converge in all our instances. **(ii)** DEC is practical and it is much more efficient than QMM.

References

[1] Megasthenis Asteris, Anastasios Kyrillidis, Oluwasanmi Koyejo, and Russell Poldrack. A simple and provable algorithm for sparse diagonal c-ca. In *International Conference on Machine Learning (ICML)*, pages 1148–1157, 2016. 1

[2] Megasthenis Asteris, Dimitris S. Papailiopoulos, and Alexandros G. Dimakis. Nonnegative sparse PCA with provable guarantees. In *International Conference on Machine Learning (ICML)*, pages 1728–1736, 2014. 6

[3] Amir Beck and Yonina C Eldar. Sparsity constrained nonlinear optimization: Optimality conditions and algorithms. *SIAM Journal on Optimization*, 23(3):1480–1509, 2013. 2, 7

[4] Amir Beck and Marc Teboulle. On minimizing quadratically constrained ratio of two quadratic functions. *Journal of Convex Analysis*, 17(3):789–804, 2010. 1

[5] Amir Beck and Yakov Vaisbourd. The sparse principal component analysis problem: Optimality conditions and algorithms. *Journal of Optimization Theory and Applications*, 170(1):119–143, 2016. 2, 6, 7

[6] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004. 5

[7] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011. 3

[8] Alexandre dAspremont, Francis Bach, and Laurent El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9(Jul):1269–1294, 2008. 2

[9] Alexandre d'Aspremont, Laurent E Ghaoui, Michael I Jordan, and Gert R Lanckriet. A direct formulation for sparse pca using semidefinite programming. In *Advances in Neural Information Processing Systems (NIPS)*, pages 41–48, 2005. 1, 2

[10] Werner Dinkelbach. On nonlinear fractional programming. *Management Science*, 13(7):492–498, 1967. 4, 5

[11] Rong Ge, Chi Jin, Praneeth Netrapalli, Aaron Sidford, et al. Efficient algorithms for large-scale generalized eigenvector computation and canonical correlation analysis. In *International Conference on Machine Learning (ICML)*, pages 2741–2750, 2016. 1

[12] Yue-Fei Guo, Shijin Li, Jing-Yu Yang, Ting-Ting Shu, and Lide Wu. A generalized foley-sammon transform based on generalized fisher discriminant criterion and its application to face recognition. *Pattern Recognition Letters*, 24(1-3):147–158, 2003. 5

[13] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 1990. 1, 5

[14] Cho-Jui Hsieh, Kai-Wei Chang, Chih-Jen Lin, S. Sathiya Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear SVM. In *International Conference on Machine Learning (ICML)*, pages 408–415, 2008. 5

[15] Cho-Jui Hsieh and Inderjit S. Dhillon. Fast coordinate descent methods with variable selection for nonnegative matrix factorization. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1064–1072, 2011. 5

[16] JNR Jeffers. Two case studies in the application of principal component analysis. *Applied Statistics*, pages 225–236, 1967. 3

[17] Thorsten Joachims. Making large-scale svm learning practical. Technical report, Technical Report, SFB 475: Komplexitätsreduktion in Multivariaten Datenstrukturen, Universität Dortmund, 1998. 3

[18] Ian T Jolliffe, Nickolay T Trendafilov, and Mudassir Uddin. A modified principal component technique based on the lasso. *Journal of computational and Graphical Statistics*, 12(3):531–547, 2003. 1

[19] Michel Journée, Yurii Nesterov, Peter Richtárik, and Rodolphe Sepulchre. Generalized power method for sparse principal component analysis. *Journal of Machine Learning Research*, 11(Feb):517–553, 2010. 2

[20] Robert Krauthgamer, Boaz Nadler, Dan Vilenchik, et al. Do semidefinite relaxations solve sparse pca up to the information limit? *The Annals of Statistics*, 43(3):1300–1322, 2015. 2

[21] Junseok Kwon and Kyoung Mu Lee. Visual tracking decomposition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 1

[22] Qi Lei, Kai Zhong, and Inderjit S. Dhillon. Coordinate-wise power method. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2056–2064, 2016. 5

[23] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 19(10):2756–2779, 2007. 6

[24] Lester W Mackey. Deflation methods for sparse pca. In *Advances in Neural Information Processing Systems*, pages 1017–1024, 2009. 3

[25] Bernard Moghaddam, Yair Weiss, and Shai Avidan. Fast pixel/part selection with sparse eigenvectors. In *IEEE International Conference on Computer Vision (ICCV)*, 2007. 1

[26] Nikhil Naikal, Allen Y. Yang, and Shankar Sastry. Informative feature selection for object recognition via a sparse PCA. In *IEEE International Conference on Computer Vision (ICCV)*, 2011. 1

[27] Sakrapee Paisitkriangkrai, Chunhua Shen, and Jian Zhang. Efficiently training a better visual detector with sparse eigenvectors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 1

[28] Sakrapee Paisitkriangkrai, Chunhua Shen, and Jian Zhang. Incremental training of a detector using online sparse eigendecomposition. *IEEE Transactions on Image Processing (TIP)*, 2011. 1

[29] Imre Pólik and Tamás Terlaky. A survey of the s-lemma. *SIAM Review*, 49(3):371–418, July 2007. 5

[30] M. J. D. Powell. On search directions for minimization algorithms. *Mathematical Programming*, 4(1):193–201, Dec 1973. 6

[31] Chunhua Shen, Sakrapee Paisitkriangkrai, and Jian Zhang. Efficiently learning a detection cascade with sparse eigenvectors. *IEEE Transactions on Image Processing (TIP)*, 2011. 1

[32] Junxiao Song, Prabhu Babu, and Daniel Pérez Palomar. Sparse generalized eigenvalue problem via smooth optimization. *IEEE Transactions on Signal Processing*, 63(7):1627–1642, 2015. 1, 7

[33] Bharath K Sriperumbudur, David A Torres, and Gert RG Lanckriet. A majorization-minimization approach to the sparse generalized eigenvalue problem. *Machine learning*, 85(1):3–39, 2011. 2

[34] Bharath K. Sriperumbudur, David A. Torres, and Gert R. G. Lanckriet. Sparse eigen methods by d.c. programming. In *International Conference on Machine Learning (ICML)*, pages 831–838, 2007. 2, 7

[35] Kean Ming Tan, Zhaoran Wang, Han Liu, and Tong Zhang. Sparse generalized eigenvalue problem: Optimal statistical rates via truncated rayleigh flow. *Journal of the Royal Statistical Society: Series B*, 2018. 1, 7

[36] Mamadou Thiao, Pham D Tao, and Le An. A dc programming approach for sparse eigenvalue problem. In *International Conference on Machine Learning (ICML)*, pages 1063–1070, 2010. 2, 7

[37] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109(3):475–494, Jun 2001. 5, 6

[38] Arnaud Vandaele, Nicolas Gillis, Qi Lei, Kai Zhong, and Inderjit S. Dhillon. Efficient and non-convex coordinate descent for symmetric nonnegative matrix factorization. *IEEE Transactions on Signal Processing*, 64(21):5571–5584, 2016. 5, 6

[39] Huan Wang, Shuicheng Yan, Dong Xu, Xiaoou Tang, and Thomas Huang. Trace ratio vs. ratio trace for dimensionality reduction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 5

[40] Ganzhao Yuan, Li Shen, and Wei-Shi Zheng. A hybrid method of combinatorial search and coordinate descent for discrete optimization. *arXiv preprint*, 2017. 2, 3, 4

[41] Xiao-Tong Yuan and Tong Zhang. Truncated power method for sparse eigenvalue problems. *Journal of*

Machine Learning Research, 14(Apr):899–925, 2013. 1, 2, 7

[42] Ailing Zhang and Shunsuke Hayashi. Celis-dennis-tapia based approach to quadratic fractional programming problems with two quadratic constraints. *Numerical Algebra, Control and Optimization*, 1(1):83–98, 2011. 4

[43] Youwei Zhang, Alexandre dAspremont, and Laurent El Ghaoui. Sparse pca: Convex relaxations, algorithms and applications. In *Handbook on Semidefinite, Conic and Polynomial Optimization*, pages 915–940. Springer, 2012. 2

[44] Youwei Zhang and Laurent E Ghaoui. Large-scale sparse principal component analysis with application to text data. In *Advances in Neural Information Processing Systems (NIPS)*, 2011. 1

[45] Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006. 1

Appendix

A. Proof of Proposition 1

Proposition 1. *When the cyclic order strategy is used, co-ordinate descent method is guaranteed to converge to a coordinate-wise minimum of Problem (10) that $\forall i$, $\mathbf{y}_i^* = \arg\min_{\alpha \geq \hat{L}} \mathcal{L}(\mathbf{y}_i^* + \alpha \mathbf{e}_i)$.*

Proof. Note that $\mathcal{L}(\mathbf{y})$ is continuous and $\{\mathcal{L}(\mathbf{y}^j)\}$ converges monotonically. Assuming that it converges to $\mathcal{L}^*$ with $\lim_{j \to \infty} \mathcal{L}(\mathbf{y}^j) = \mathcal{L}^*$, we obtain that $\forall \alpha$, $i = 1, .., m$:

$$\mathcal{L}^* = \mathcal{L}(\mathbf{y}^{j-1}) = \mathcal{L}(\mathbf{y}^j) \leq \mathcal{L}(\mathbf{y}^{j-1} + \alpha \mathbf{e}_i). \quad (12)$$

Therefore, the right-handed side in (12) attains it minimum at both 0 and $(\mathbf{y}^j)_i - (\mathbf{y}^{j-1})_i$. Combining with the fact that the subproblem only contains one unique global solution, we have $(\mathbf{y}^{j-1})_i = (\mathbf{y}^j)_i$. Since the coordinate i is picked using cyclic order, we have: $\mathbf{y}^{j-1} = \mathbf{y}^j = \mathbf{y}^*$ and $\mathbf{y}^*$ is a coordinate-wise minimum point. $\square$

B. Proof of Lemma 2

Lemma 2. *(**Sufficient Decrease Condition**) It holds that:*
$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \leq \frac{-\theta \|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2}{(\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1}}$.

Proof. We let B be the working set in the t-th iteration and $N \triangleq \{1, 2, ..., n\} \setminus B$. Since we solve Problem (3) in the t-th iteration, we have:

$$(h(\mathbf{x}_B^{t+1}, \mathbf{x}_N^t) + \frac{\theta}{2}\|\mathbf{x}_B^{t+1} - \mathbf{x}_B^t\|_2^2) / g(\mathbf{x}_B^{t+1}, \mathbf{x}_N^t)$$
$$\leq (h(\mathbf{z}, \mathbf{x}_N^t) + \frac{\theta}{2}\|\mathbf{z} - \mathbf{x}_B^t\|_2^2) / g(\mathbf{z}, \mathbf{x}_N^t), \ \forall \mathbf{z} \in \mathbb{R}^k.$$

We let $\mathbf{z} = \mathbf{x}_B^t$ and combine with the fact that $\mathbf{x}_N^{t+1} = \mathbf{x}_N^t$, we have:

$$(h(\mathbf{x}_B^{t+1}, \mathbf{x}_N^{t+1}) + \frac{\theta}{2}\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2) / g(\mathbf{x}_B^{t+1}, \mathbf{x}_N^{t+1})$$
$$\leq (h(\mathbf{x}_B^t, \mathbf{x}_N^t) + 0) / g(\mathbf{x}_B^t, \mathbf{x}_N^t).$$

Noticing the fact that $h(\mathbf{x}_B^t, \mathbf{x}_N^t) = \frac{1}{2}(\mathbf{x}^t)^T \mathbf{A} \mathbf{x}^t$ and $g(\mathbf{x}_B^t, \mathbf{x}_N^t) = \frac{1}{2}(\mathbf{x}^t)^T \mathbf{C} \mathbf{x}^t$, we have:

$$((\mathbf{x}^{t+1})^T \mathbf{A} \mathbf{x}^{t+1} + \theta\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2) / ((\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1})$$
$$\leq ((\mathbf{x}^t)^T \mathbf{A} \mathbf{x}^t) / ((\mathbf{x}^t)^T \mathbf{C} \mathbf{x}^t).$$

Moreover, using the structure of the objective function $f(\cdot)$, we obtain: $f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) = \frac{(\mathbf{x}^{t+1})^T \mathbf{A} \mathbf{x}^{t+1}}{(\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1}} - \frac{(\mathbf{x}^t)^T \mathbf{A} \mathbf{x}^t}{(\mathbf{x}^t)^T \mathbf{C} \mathbf{x}^t} \leq \frac{-\theta\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2}{(\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1}}$. Thus, we finish the proof of this lemma.
$\square$

C. Proof of Theorem 2

We now prove the convergence properties of Algorithm 1. The following supermartingale convergence result is useful in our analysis [31].

Lemma 3. *[31] Let $\mathbf{v}_t$, $\mathbf{u}_t$ and $\boldsymbol{\alpha}_t$ be three sequences of nonnegative random variables such that*

$$\mathbb{E}[\mathbf{v}_{t+1} \mid \mathcal{F}_t] \leq (1 + \boldsymbol{\alpha}_t)\mathbf{v}_t - \mathbf{u}_t, \ \forall t \geq 0 \ a.s.$$
$$\text{and } \textstyle\sum_{t=0}^{\infty} \boldsymbol{\alpha}_t < \infty \ a.s., \quad (13)$$

where $\mathcal{F}_t$ denotes the collections $\{\mathbf{v}_0, ..., \mathbf{v}_t,\ \mathbf{u}_0, ..., \mathbf{u}_t, \boldsymbol{\alpha}_0, ..., \boldsymbol{\alpha}_t\}$. Then, we have $\lim_{t \to \infty} \mathbf{v}_t = \chi$ for a random variable $\chi \geq 0$ a.s. and $\sum_{t=0}^{\infty} \mathbf{u}_t < \infty$ a.s.

We now present our main results.

Theorem 2. ***Convergence Properties of Algorithm 1.*** *Assume that the subproblem in (3) is solved globally, and there exists a constant σ such that $\mathbf{x}^t \mathbf{C} \mathbf{x}^t \geq \sigma > 0$ for all t. We have the following results.*

(i) When the random strategy is used to find the working set, we have $\lim_{t \to \infty} \mathbb{E}[\|\mathbf{x}^{t+1} - \mathbf{x}^t\|] = 0$ and Algorithm 1 converges to the block-k stationary point in expectation.

(ii) When the swapping strategy is used to find the working set with $k \geq 2$, we have $\lim_{t \to \infty} \|\mathbf{x}^{t+1} - \mathbf{x}^t\| = 0$ and Algorithm 1 converges to the block-2 stationary point deterministically.

Proof. We use $\mathbf{x}^*$ and $\bar{\mathbf{x}}$ to denote any optimal point and any block-k stationary point of (1), respectively. We use the notation ξ^t for the entire history of random index selection:

$$\xi^t = \{B^0, \ B^1, \ ..., B^t\}$$

(i) We notice that B^t is independent on the past B^{t-1}, while $\mathbf{x}^t$ fully depends on ξ^{t-1}. Taking the expectation conditioned on ξ^{t-1} for the sufficient descent inequality in Lemma 2, we obtain:

$$\mathbb{E}[f(\mathbf{x}^{t+1})|\xi^t] - f(\mathbf{x}^t)$$
$$\leq -\mathbb{E}[\frac{\theta\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2}{(\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1}}|\xi^t]$$
$$\overset{(a)}{\leq} -\frac{\theta}{\sigma}\mathbb{E}[\|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2|\xi^t]$$
$$= -\frac{\theta}{\sigma}\frac{1}{C_n^k}\textstyle\sum_{i=1}^{C_n^k} \|\mathcal{P}(\mathcal{B}_{(i)}, \mathbf{x}^t) - \mathbf{x}_{\mathcal{B}_{(i)}}^t\|_2^2$$
$$\overset{(b)}{=} -\frac{\theta}{\sigma} \cdot \mathcal{M}(\mathbf{x}^t) \quad (14)$$

step (a) uses the assumption that $\mathbf{x}^t \mathbf{C} \mathbf{x}^t \geq \sigma > 0$, $\forall \mathbf{x}^t$ which clearly holds since $\mathbf{C}$ is strictly positive and $\mathbf{x}^t \neq \mathbf{0}$; step (b) uses the definition of $\mathcal{M}(\mathbf{x}^t)$ in Definition 1. Therefore, we have:

$$\mathbb{E}[f(\mathbf{x}^{t+1}) \mid \xi^t] - f(\mathbf{x}^*) \leq f(\mathbf{x}^t) - f(\mathbf{x}^*) - \frac{\theta \mathcal{M}(\mathbf{x}^t)}{\sigma} \quad (15)$$
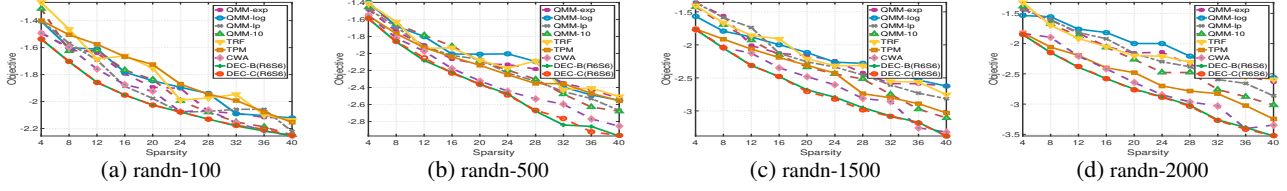
(a) randn-100   (b) randn-500   (c) randn-1500   (d) randn-2000

Figure 6 Accuracy of different methods on different data sets for sparse PCA problem with varying the cardinalities.

(a) randn-100   (b) randn-500   (c) randn-1500   (d) randn-2000
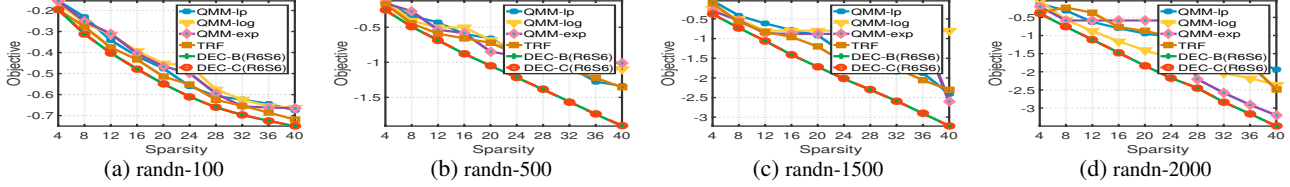
Figure 7 Accuracy of different methods on different data sets for sparse FDA problem with varying the cardinalities.

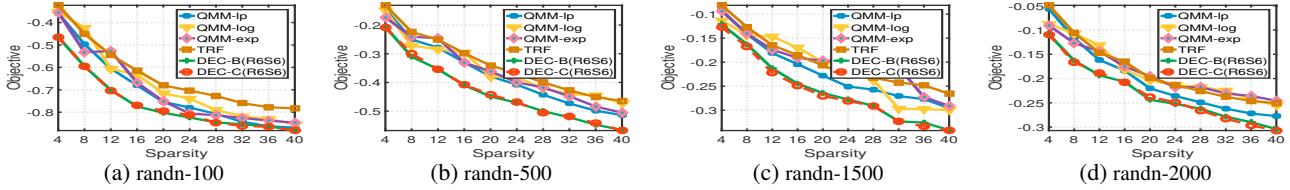(a) randn-100   (b) randn-500   (c) randn-1500   (d) randn-2000

Figure 8 Accuracy of different methods on different data sets for sparse CCA problem with varying the cardinalities.

Using the supermartingale convergence theorem given in Lemma 3 with $\mathbf{v}_t = \mathbb{E}[f(\mathbf{x}^{t+1}) \mid \xi^t] - f(\mathbf{x}^*) \geq 0$ and $\mathbf{u}_t = \frac{\theta \mathcal{M}(\mathbf{x}^t)}{\sigma}$, we have

$$\lim_{t \to \infty} f(\mathbf{x}^t) - f(\mathbf{x}^*) = \chi \ a.s.$$

for a certain random variable $\chi \geq 0$ and thus the sequence $f(\mathbf{x}^t)$ converges to a random variable $\bar{F} = \chi + f(\mathbf{x}^*)$. In addition, we have $\lim_{t \to \infty} f(\mathbf{x}^t) - f(\mathbf{x}^{t+1}) = 0$ almost surely. From (14), we have

$$\lim_{t \to \infty} \mathcal{M}(\mathbf{x}^t) = 0, \quad \lim_{t \to \infty} \|\mathbf{x}^t - \mathbf{x}^{t+1}\| = 0.$$

Therefore, the algorithm converges to the block-k stationary point. Summing the inequality in (14) over $i = 0, 1, ..., t - 1$, we have:

$$\frac{\theta}{\sigma} \cdot \sum_{i=0}^{t} \mathcal{M}(\mathbf{x}^i) \leq f(\mathbf{x}^0) - f(\mathbf{x}^t).$$

Using the fact that $f(\mathbf{x}^*) \leq f(\mathbf{x}^t)$, we obtain:

$$\frac{\theta}{\sigma} \sum_{i=0}^{t} \mathbb{E}[\|\mathcal{M}(\mathbf{x}^i) \mid \xi^i] \leq f(\mathbf{x}^0) - f(\mathbf{x}^*)$$

$$\Rightarrow \quad \min_{i=1,...,t} \mathbb{E}[\mathcal{M}(\mathbf{x}^i) \mid \xi^i] \leq \frac{\sigma(f(\mathbf{x}^0) - f(\mathbf{x}^*))}{t\theta}.$$

We conclude that $\mathbf{x}^t$ converges to the block-k stationary point with $\min_{i=1,...,t} \mathbb{E}[\mathcal{M}(\mathbf{x}^i) \mid \mathbf{x}^i] \leq \mathcal{O}(1/t)$.

(ii) We now prove the second part of this theorem. We have the following inequalities:

$$\begin{aligned} f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) &\leq -\frac{\theta \|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2}{(\mathbf{x}^{t+1})^T \mathbf{C} \mathbf{x}^{t+1}} \\ &\leq -\frac{\theta}{\sigma} \|\mathbf{x}^{t+1} - \mathbf{x}^t\|_2^2 \end{aligned}$$

Summing this inequality over $i = 0, 1, ..., t - 1$, we have:

$$\frac{\theta}{\sigma} \cdot \sum_{i=0}^{t} \|\mathbf{x}^{i+1} - \mathbf{x}^i\|_2^2 \leq f(\mathbf{x}^0) - f(\mathbf{x}^t)$$

$$\Rightarrow \quad \min_{i=1,...,t} \|\mathbf{x}^{i+1} - \mathbf{x}^i\|_2^2 \leq \frac{\sigma}{\theta} \frac{f(\mathbf{x}^0) - f(\mathbf{x}^*)}{t}.$$

Using the fact that $f(\mathbf{x}^*) \leq f(\mathbf{x}^t)$, we have $\lim_{t \to \infty} \|\mathbf{x}^{t+1} - \mathbf{x}^t\| = 0$. Therefore, Algorithm 1 is convergent when swapping strategy is used.

We now prove that Algorithm 1 convergence to a block-2 stationary point $\bar{\mathbf{x}}$. Since Algorithm 1 is monotonically non-increasing and converges to a stationary point $\bar{\mathbf{x}}$ such that no decrease is made, we have $\mathbf{D}_{i,j} \geq 0$ for (4). Therefore, it holds that $\min_\alpha \ f(\bar{\mathbf{x}} + \alpha \mathbf{e}_i - (\bar{\mathbf{x}})_j \mathbf{e}_j) \geq f(\bar{\mathbf{x}}), \ \forall i \in \bar{\mathcal{S}}(\bar{\mathbf{x}}), \ j \in \bar{\mathcal{Z}}(\bar{\mathbf{x}})$. We have the following result: $f(\bar{\mathbf{x}}) \leq f(\bar{\mathbf{x}} + \mathbf{d}), \forall \mathbf{d}$ with $\|\mathbf{d} - \bar{\mathbf{x}}\|_0 = 2$. Therefore, $\bar{\mathbf{x}}$ is a block-2 stationary point. $\qquad \square$

D. Additional Experiments

We demonstrate the experimental results on the randomized generated data sets for sparse PCA, sparse FDA, and sparse CCA in Figure 6, 7 and 8, respectively. These results further consolidate our conclusions drawn in Section 7.