New Formulation and Computation for Generalized Singular Values of Grassman Matrix Pair

Wei-Wei Xu∗ Michael K. Ng† Zheng-Jian Bai‡

April 7, 2020

Abstract

In this paper, we derive new model formulations for computing generalized singular values of a Grassman matrix pair. These new formulations make use of truncated filter matrices to locate the $i$-th generalized singular value of a Grassman matrix pair. The resulting matrix optimization problems can be solved by using numerical methods involving Newton’s method on Grassmann manifold. Numerical examples on synthetic data sets and gene expression data sets are reported to demonstrate the high accuracy and the fast computation of the proposed new formulations for computing arbitrary generalized singular value of Grassman matrix pair.

Keywords. New model formulation, arbitrary generalized singular value, Grassman matrix pair, Newton’s method on Grassmann manifold

2010 AMS subject classifications. 65F15, 65F99.

1 Introduction

The generalized singular value decomposition (GSVD) used in mathematics and numerical computations [10, 11, 22] is a very useful and versatile tool. The GSVD of two matrices having the same number of columns was first proposed by Van Loan [22]. It is very useful in many matrix computation problems and practical applications, such as the Kronecker canonical form of a general matrix pencil, the linearly constrained least-squares problem, the general Gauss-Markov

∗School of Mathematics and Statistics, Nanjing University of Information Science and Technology, Nanjing 210044, People’s Republic of China (wwx19840904@sina.com). The research of this author is partially supported by the National Natural Science Foundation of China (No. 11971243) and the Natural Science Foundation of Jiangsu Province of China (No. BK20181405).
†Department of Mathematics, The University of Hong Kong, Pokfulam, Hong Kong, People’s Republic of China (mng@maths.hku.hk). The research of this author is partially supported by the HKRGC GRF 12306616, 12200318, 12300519, and HKU 104005583.
‡School of Mathematical Sciences and Fujian Provincial Key Laboratory on Mathematical Modeling & High Performance Scientific Computing, Xiamen University, Xiamen 361005, People’s Republic of China (zjbai@xmu.edu.cn). The research of this author is partially supported by the National Natural Science Foundation of China (No. 11671337) and the Fundamental Research Funds for the Central Universities (No. 20720180008).
linear model, the generalized total least-squares problem, real-time signal processing, comparative analysis of DNA microarrays and so on; see for example [2, 3, 4, 5, 6, 7, 8, 10, 11, 14, 18, 20, 22, 23, 24, 26].

Numerical methods and perturbation analysis of GSVD have been well developed; see for instance [4, 7, 8, 14, 18, 20, 21, 22, 23, 24, 26]. The GSVD of two matrices having the same number of columns was first proposed by Van Loan [22]. Van Loan [22] and Paige [18] provided algorithms for computing the generalized singular value decomposition. Bai and Demmel [4] described a variation of Paige’s algorithm for computing the GSVD introduced by Van Loan [22] and Paige and Saunders [19]. Ewerbring and Luk [8] and Zha [26] proposed a GSVD for matrix triplets. Stewart [20] and Van Loan [23] proposed two algorithms for computing the GSVD. On perturbation analysis of the GSVD, Sun [21] and Li [14] presented several perturbation bounds of generalized singular values (GSVs) of a Grassman matrix pair (GMP) and their associated subspaces. Xu et al. [24] provided the explicit expression and sharper bounds of the chordal metric between GSVs of a GMP.

Recently, the GSVD plays an important role in the analysis of DNA microarrays and gene data. For arbitrary generalized singular value \((\alpha_i, \beta_i)\) of a GMP, the ratio \(\alpha_i/\beta_i\) can be used in comparison analysis of gene data as an indicator; see for example [2, 3]. Moreover, the formulation for arbitrary GSV of a GMP has not been studied. Usually, the GSVD are computed by making decompositions (e.g., QR decomposition and CS decomposition). In these cases, computing the whole decomposition makes computational cost higher. The main goal of this paper is to propose new model formulations for computing arbitrary GSV of a GMP. We first derive new formulations for computing arbitrary GSV of the GMP. By using truncated filter matrices, the matrix optimization problems can be reformulated to locate the \(i\)-th GSV of the GMP. The resulting optimization problems can be solved by using Newton’s method on Grassmann manifold. Numerical examples on synthetic data sets and gene expression data sets are reported to demonstrate the high accuracy and the fast computation of the proposed method for computing arbitrary GSV of a GMP.

1.1 Organization

The rest of this paper is organized as follows. In Section 2, we provide mathematical preliminaries and derive new formulations for computing GSVs of a GMP. In Section 3, we present the numerical method for solving the resulting optimization models. In Section 4, we provide numerical examples to show the efficiency of the theoretical results. Finally, some concluding remarks are given in Section 5.

1.2 Notation

Throughout this paper we always use the following notation and definitions. \(i\) denotes imaginary unit \(\sqrt{-1}\) and \(\mathbb{R}, \mathbb{C}, \mathbb{R}^n, \mathbb{C}^{m \times n}\) and \(\mathbb{U}_n\) are the sets of real numbers, complex numbers, \(n\)-dimensional real vectors, \(m \times n\) complex matrices and \(n \times n\) unitary matrices accordingly. \(|\cdot|\) stands for the absolute value of a complex number. \(I_n\) and \(O_{m \times n}\) stand for the identity matrix of order \(n\) and the \(m \times n\) zero matrix, respectively. \(A, A^T, A^H, A^{-1}, \det(A), \text{tr}(A)\) denote the conjugate, transpose, conjugate transpose, inverse, determinant and trace of a matrix \(A\).
accordingly. By \( \| \cdot \|_2 \) we denote the spectral norm of a matrix. The singular value set of \( A \) is denoted by \( \sigma(A) \). For given matrices \( A, B \in \mathbb{C}^{n \times n} \), \( A < (\leq) B \) means \( B - A \) is a positive (semi-)definite matrix. For a matrix \( A \in \mathbb{C}^{n \times n} \), we denote by \( \sigma_1(A) \geq \sigma_2(A) \geq \cdots \geq \sigma_n(A) \geq 0 \) its singular values, arranged in decreasing order. We denote [\( \gamma \)] the largest integer less than or equal to a real number \( \gamma \). The symbol “\( \otimes \)” means the Kronecker product and vec(\( \cdot \)) creates a column vector from a matrix by stacking its column vectors below one another. Finally, diag(\( A, B \)) denotes a block diagonal matrix in which the diagonal blocks are square matrices \( A \) and \( B \).

**Definition 1.1** [10] Let \( A \in \mathbb{C}^{m \times n} \) and \( B \in \mathbb{C}^{p \times n} \). A matrix pair \( \{A, B\} \) is an \((m, p, n)\)-GMP if rank \((A^T, B^T)^T = n\).

For any \((m, p, n)\)-GMP \( \{A, B\} \), one may see \( Z = (A^T, B^T)^T \) as a point of the complex projective space \( G_{n,m+p} \) of all \( n \)-dimensional subspaces of the \((m+p)\)-dimensional complex space \( \mathbb{C}^{m+p} \), i.e., one can identify \( Z \) with the linear subspace \( R(Z) \) (see e.g. [15 17 21]). Clearly, if \( \{A, B\} \) is an \((m, p, n)\)-GMP, then \((A^HA, B^HB)\) is a definite matrix pair, i.e., \( x^HA^HAx + x^HB^HBx > 0 \) for all nonzero \( x \in \mathbb{C}^n \). The definite pair \((A^HA, B^HB)\) has \( n \) generalized eigenvalues, and thus the GMP \( \{A, B\} \) has \( n \) generalized singular values. A well-developed perturbation theory for the generalized eigenvalue problem of definite pencils is available. Perturbation bounds for the generalized singular value problem can be obtained with the help of the close relation between the two problems. However, the bounds obtained in this way are often unsatisfactory, just like the perturbation bounds for the singular values (SVs) of a single matrix \( A \) obtained through the perturbation bounds for the eigenvalues of the Hermitian matrix \( A^HA \). Therefore, special attention deserves to be paid to perturbations for the generalized singular value problem. Therefore, research on numerical methods and perturbation analysis of the GSVD of a GMP is an important topic (see e.g. [3 14 21 22 24]).

**Definition 1.2** [10] Let \( \{A, B\} \) be an \((m, p, n)\)-GMP. A nonnegative number pair \((\alpha, \beta)\) is a GSV of the GMP \( \{A, B\} \) if

\[
(\alpha, \beta) \neq (0, 0), \quad \det(\beta^2 A^H A - \alpha^2 B^H B) = 0, \quad \alpha, \beta \geq 0.
\]

The set of GSV of \( \{A, B\} \) is denoted by \( \sigma\{A, B\} \). Evidently,

\[
\sigma\{A, B\} = \{(\alpha, \beta) \neq (0, 0) | \det(\beta^2 A^H A - \alpha^2 B^H B) = 0, \alpha, \beta \geq 0\}.
\]

In the literature [22 23], there are several formulations of the GSVD. Here we adopt the following form.

**Definition 1.3** Let \( \{A, B\} \) be an \((m, p, n)\)-GMP. Then there exist unitary matrices \( U \in \mathbb{C}^{m \times m}, V \in \mathbb{C}^{p \times p} \), and a nonsingular matrix \( R \in \mathbb{C}^{n \times n} \) such that

\[
U^H A R^{-1} = \Sigma_A, \quad V^H B R^{-1} = \Sigma_B, \quad \tag{1.1}
\]

\[
\Sigma_A = \begin{pmatrix} \Lambda & \mathbf{0}_{(m-r-s) \times (n-r-s)} \\ \mathbf{0}_{(m-r-s) \times (n-r-s)} & \mathbf{0}_{(p+r-n) \times r} \end{pmatrix}, \quad \Sigma_B = \begin{pmatrix} \mathbf{0}_{(p+r-n) \times r} & \Omega \end{pmatrix}, \quad \tag{1.2}
\]

where

\[
\Lambda = \text{diag}(\alpha_1, \ldots, \alpha_{r+s}), \quad \Omega = \text{diag}(\beta_{r+1}, \ldots, \beta_n),
\]
with
\[ 1 = \alpha_1 = \cdots = \alpha_r > \alpha_{r+1} \geq \cdots \geq \alpha_{r+s} > \alpha_{r+s+1} = \cdots = \alpha_n = 0, \]
\[ 0 = \beta_1 = \cdots = \beta_r < \beta_{r+1} \leq \cdots \leq \beta_{r+s} < \beta_{r+s+1} = \cdots = \beta_n = 1, \]
and
\[ \alpha_i^2 + \beta_i^2 = 1, \quad 1 \leq i \leq n. \]

2 Trace function optimization model formulation

In this section we give new model formulation of any GSV of a GMP by trace function under one variable.

**Lemma 2.1** \[25\] Let \( c \in \mathbb{R} \) and \( A_1, \ldots, A_m \in \mathbb{C}^{n \times n} \) with \( \sigma_1(A_i) \geq \sigma_2(A_i) \geq \cdots \geq \sigma_n(A_i) \) for \( i = 1, \ldots, m \). Then we have

\[
\max_{U_1, \ldots, U_m \in 
\mathbb{U}} \Re[\text{tr}(cI_n \pm \prod_{j=1}^{m} U_j A_j)] = nc + \sum_{i=1}^{n} \prod_{j=1}^{m} \sigma_i(A_j),
\]

\[
\max_{U_1, \ldots, U_m \in 
\mathbb{U}} [\text{tr}(cI_n \pm \prod_{j=1}^{m} U_j A_j)] = n |c| + \sum_{i=1}^{n} \prod_{j=1}^{m} \sigma_i(A_j),
\]

\[
\min_{U_1, \ldots, U_m \in 
\mathbb{U}} \Re[\text{tr}(cI_n \pm \prod_{j=1}^{m} U_j A_j)] = \begin{cases} 
nc - \sum_{i=1}^{n} \prod_{j=1}^{m} \sigma_i(A_j), & \text{if } \frac{1}{n} \sum_{i=1}^{n} \prod_{j=1}^{m} \sigma_i(A_j) \leq |c|, \\
0, & \text{if } \frac{1}{n} \sum_{i=1}^{n} \prod_{j=1}^{m} \sigma_i(A_j) \geq |c|. 
\end{cases}
\]

**Lemma 2.2** \[12\] Let \( f(X) : \mathbb{C}^{n \times n} \to \mathbb{R} \) be an analytical function of several complex variables on the domain \( XX^H \leq I_n \). Then \( f(X) \) attains its maximum modulus on the characteristic manifold \( \{ X \in \mathbb{C}^{n \times n} : XX^H = I_n \} \).

We now give new formula model of any GSV of a GMP by trace function optimization under one variable.

**Lemma 2.3** Let \( \{ A, B \} \) be an \((m,p,n)\)-GMP and the GSV \((\alpha_i, \beta_i)\) of \( \{ A, B \} \) be given in (1.1) and (1.2). The following conclusions hold true.

a) If \( n \geq m \), then for \( 1 \leq i \leq n \),

\[
\alpha_i^2 = \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(A^H \Phi_1^H Q_i \Phi_1 A(A^H A + B^H B)^{-1}) \]
\[ - \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(A^H \Phi_1^H Q_{i-1} \Phi_1 A(A^H A + B^H B)^{-1}) = \phi_i, \quad (2.1) \]

where
\[
Q_i = \text{diag}(T_i, O_{(m-n) \times (m-n)}) \quad \text{with} \quad T_i = \text{diag}(I_i, O_{(n-i) \times (n-i)}). \quad (2.2)
\]
b) If \( n \leq p \), then for \( 1 \leq i \leq n \),

\[
\beta_i^2 = \max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1}) - \max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1}) \equiv \psi_i,
\]

(2.3)

where

\[
P_i = \text{diag}(O_{(p-n) \times (p-n)}, F_i)
\]

with \( F_i = \text{diag}(O_{(i-1) \times (i-1)}, I_{n-i+1}) \).

(2.4)

c) If \( m \leq n \), then \( \alpha_{m+1} = \cdots = \alpha_n = 0 \) and for \( 1 \leq i \leq m \),

\[
\alpha_i^2 = \max_{\Psi_i \in \mathbb{U}_m} \text{tr}(A^H \Psi_i^H P_i \Psi_i A (A^H A + B^H B)^{-1}) - \max_{\Psi_i \in \mathbb{U}_m} \text{tr}(A^H \Psi_i^H P_i \Phi_i A (A^H A + B^H B)^{-1}) \equiv \varphi_i,
\]

(2.5)

where

\[
Q_i = \text{diag}(I_i, O_{(m-i) \times (m-i)}).
\]

(2.6)

d) If \( p \leq n \), then \( \beta_1 = \cdots = \beta_{n-p} = 0 \) and for \( n-p+1 \leq i \leq n \),

\[
\beta_i^2 = \max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1}) - \max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1}) \equiv \chi_i.
\]

(2.7)

where

\[
P_i = \text{diag}(O_{(p-n-i+1) \times (p-n-i+1)}, I_{n-i+1}).
\]

(2.8)

**Proof:**

a) If \( n \leq m \), then for \( 1 \leq i \leq n \), we let \( Q_i \) be defined by (2.2). We set \( \Sigma_A = \left( \begin{smallmatrix} \hat{\Sigma}_A \\ O_{(m-n) \times n} \end{smallmatrix} \right) \) and for any \( \Phi_1 \in \mathbb{U}_m \), we let \( \Phi_1 U = \begin{pmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{pmatrix} \) with \( \phi_{11} \in \mathbb{C}^{n \times n} \). Using the GSVD of \( \{A, B\} \) in (1.1) and (1.2) we have

\[
\max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(A^H \Phi_1^H Q_i \Phi_1 A (A^H A + B^H B)^{-1}) = \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}((A^H A + B^H B)^{-1/2} A^H \Phi_1^H Q_i \Phi_1 A (A^H A + B^H B)^{-1/2})
\]

\[
= \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(\Sigma_A^H U^H \Phi_1^H Q_i \Phi_1 U \Sigma_A)
\]

\[
= \max_{\phi_{11} \Phi_{11}^H \leq I_n} \text{tr}(\Sigma_A^H \phi_{11}^H T_i \phi_{11} \hat{\Sigma}_A).
\]

Observe \( f(\phi_{11}) = \text{tr}(\hat{\Sigma}_A^H \phi_{11}^H T_i \phi_{11} \hat{\Sigma}_A) \) is an analytical function of several complex variables on the domain \( \phi_{11} \Phi_{11}^H \leq I_n \). By Lemma 2.2 we have \( f(\phi_{11}) \) attains its maximum modulus on the characteristic manifold \( \{\phi_{11} \in \mathbb{C}^{n \times n} : \phi_{11} \Phi_{11}^H = I_n\} \). Therefore,

\[
\max_{\phi_{11} \Phi_{11}^H \leq I_n} \text{tr}(\hat{\Sigma}_A^H \phi_{11}^H T_i \phi_{11} \hat{\Sigma}_A) = \max_{\phi_{11} \Phi_{11}^H = I_n} \text{tr}(\hat{\Sigma}_A^H \phi_{11}^H T_i \phi_{11} \hat{\Sigma}_A).
\]
Similarly, we have
\[\max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(A^H \Phi_1^H Q_1 \Phi_1 A (A^H A + B^H B)^{-1}) = \max_{\phi_{i1} \phi_{i1}^T = I_n} \text{tr}(\hat{\Sigma}_A^H \phi_{i1}^T T_i \phi_{i1} \hat{\Sigma}_A) = \alpha_1^2 + \cdots + \alpha_r^2.\] (2.9)

From (2.9) and (2.10) it follows that (2.1) holds for 1 \leq i \leq n.

b) If \(n \leq p\), then for 1 \leq i \leq n, let \(P_i\) be defined by (2.4). Let \(\Sigma_B = (O_{(p-n)\times n}, \hat{\Sigma}_B)\). For any \(\Phi_2 \in \mathbb{U}_p\), let \(\Phi_2 V = \begin{pmatrix} \psi_{11} & \psi_{12} \\ \psi_{21} & \psi_{22} \end{pmatrix} \in \mathbb{U}_p\) with \(\psi_{22} \in \mathbb{C}^{n \times n}\). Using the GSVD of \(\{A, B\}\) in (1.1) and (1.2) we have

\[\max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1})\]
\[= \max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(\Sigma_B^H V^H \Phi_2^H P_i \Phi_2 V \Sigma_B)\]
\[= \max_{\psi_{22} \psi_{22}^H \leq I_n} \text{tr}(\hat{\Sigma}_B^H \psi_{22}^H F_i \psi_{22} \hat{\Sigma}_B),\]

which attains its maximum on the characteristic manifold \(\{\psi_{22} \in \mathbb{C}^{n \times n} : \psi_{22} \psi_{22}^H = I_n\}\) by Lemma 2.2. Then by Lemma 2.1 we have

\[\max_{\Phi_2 \in \mathbb{U}_p} \text{tr}(B^H \Phi_2^H P_i \Phi_2 B (A^H A + B^H B)^{-1}) = \max_{\psi_{22} \psi_{22}^H \leq I_n} \text{tr}(\hat{\Sigma}_B^H \psi_{22}^H F_i \psi_{22} \hat{\Sigma}_B) = \beta_1^2 + \cdots + \beta_n^2.\] (2.11)

Similarly, we have
\[\max_{\Phi_1 \in \mathbb{U}_p} \text{tr}(B^H \Phi_1^H P_i + 1 \Phi_1 B (A^H A + B^H B)^{-1}) = \beta_{i+1}^2 + \cdots + \beta_n^2.\] (2.12)

Using (2.11) and (2.12) we can conclude that (2.3) holds for 1 \leq i \leq n.

c) If \(m \leq n\), then it is easy to check that \(\alpha_{m+1} = \cdots = \alpha_n = 0\). For 1 \leq i \leq m, let \(Q_i\) be defined by (2.6). We set \(\Sigma_A = (\hat{\Sigma}_A, O_{m \times (n-m)}).\) Using the GSVD of \(\{A, B\}\) in (1.1) and (1.2) and Lemma 2.1 we have

\[\max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(A^H \Phi_1^H Q_i \Phi_1 A (A^H A + B^H B)^{-1})\]
\[= \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(\Sigma_A^H U^H \Phi_1^H Q_i \Phi_1 U \Sigma_A)\]
\[= \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}\left(\text{diag}(\Sigma_A^H U^H \Phi_1^H Q_i \Phi_1 U \Sigma_A, O_{(n-m)\times(n-m)})\right)\]
\[= \max_{\Phi_1 \in \mathbb{U}_m} \text{tr}(\hat{\Sigma}_A^H U^H \Phi_1^H Q_i \Phi_1 U \hat{\Sigma}_A)\]
\[= \alpha_1^2 + \cdots + \alpha_i^2.\] (2.13)
Similarly, we have
\[
\max_{\Psi_1 \in \mathbb{U}_m} \text{tr}(A^H \Psi_1 H Q_{i-1} \Psi_1 A (A^H A + B^H B)^{-1}) = \alpha_1^2 + \cdots + \alpha_{i-1}^2. \tag{2.14}
\]
From (2.13) and (2.14) it follows that (2.5) holds for \(1 \leq i \leq m\).

d) If \(p \leq n\), then \(\beta_1 = \cdots = \beta_{n-p} = 0\). For \(n - p + 1 \leq i \leq n\), let \(P_i\) be defined by (2.8).

Next, we give new formula model of any GSV of a GMP by trace function optimization under two variables.

**Lemma 2.4** Let \(\{A, B\}\) be an \((m, p, n)\)-GMP and the GSV \((\alpha_i, \beta_i)\) of \(\{A, B\}\) be given in (1.1) and (1.2). The following conclusions hold true.

a) If \(n \leq m\), then for \(1 \leq i \leq n\),
\[
\alpha_i = \max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 A (A^H A + B^H B)^{-1/2} \Pi_2 G_i)|
\]
\[
- \max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 A (A^H A + B^H B)^{-1/2} \Pi_2 G_{i-1})| \equiv \tilde{\phi}_i, \tag{2.17}
\]
where
\[
G_i = (G_i, O_{n \times (m-n)}), \quad G_i = \text{diag}(I_i, O_{(n-i) \times (n-i)}).
\]

b) If \(n \leq p\), then for \(1 \leq i \leq n\),
\[
\beta_i = \max_{\Xi_1 \in \mathbb{U}_p, \Xi_2 \in \mathbb{U}_n} |\text{tr}(\Xi_1 B (A^H A + B^H B)^{-1/2} \Xi_2 H_i)|
\]
\[
- \max_{\Xi_1 \in \mathbb{U}_p, \Xi_2 \in \mathbb{U}_n} |\text{tr}(\Xi_1 B (A^H A + B^H B)^{-1/2} \Xi_2 H_{i-1})| \equiv \tilde{\psi}_i, \tag{2.19}
\]
where
\[
H_i = (O_{n \times (p-n)}, H_i), \quad H_i = \text{diag}(O_{(i-1) \times (i-1)}, I_{n-i+1}). \tag{2.20}
\]
c) If \( m \leq n \), then \( \alpha_{m+1} = \cdots = \alpha_n = 0 \) and for \( 1 \leq i \leq m \),

\[
\alpha_i = \max_{\Pi_3 \in \mathbb{U}_n, \Pi_4 \in \mathbb{U}_m} |\text{tr}(\Pi_3(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \mathbf{A}^H \Pi_4 \mathbf{S}_i)| - \max_{\Pi_3 \in \mathbb{U}_n, \Pi_4 \in \mathbb{U}_m} |\text{tr}(\Pi_3(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \mathbf{A}^H \Pi_4 \mathbf{S}_{i-1})| \equiv \hat{\varphi}_i ,
\]

(2.21)

\[
S_i = (S_i, O_{m \times (n-m)}), \quad S_i = \text{diag}(I_i, O_{(m-i) \times (m-i)}).
\]

(2.22)

d) If \( p \leq n \), then \( \beta_1 = \cdots = \beta_{n-p} = 0 \) and for \( n-p+1 \leq i \leq n \),

\[
\beta_i = \max_{\Xi_3 \in \mathbb{U}_n, \Xi_4 \in \mathbb{U}_p} |\text{tr}(\Xi_3(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \mathbf{B}^H \Xi_4 W_i)| - \max_{\Xi_3 \in \mathbb{U}_n, \Xi_4 \in \mathbb{U}_p} |\text{tr}(\Xi_3(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \mathbf{B}^H \Xi_4 W_{i-1})| \equiv \hat{\chi}_i ,
\]

(2.23)

where

\[
W_i = (O_{(p \times (n-p))}, W_i), \quad W_i = \text{diag}(O_{(p-i+1) \times (p-i+1)}, I_{n-i+1}).
\]

(2.24)

**Proof:**

a) If \( n \leq m \), then for \( 1 \leq i \leq n \), we let \( \mathcal{G}_i \) be defined by (2.18). We set \( \Sigma_A = (\hat{\Sigma}_A)
\)

\[
\hat{\Sigma}_A = (O_{(m-n) \times n}).
\]

For any \( \Pi_1 \in \mathbb{U}_m \), let \( \Pi_1 U = \begin{pmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{21} & \Pi_{22} \end{pmatrix} \) with \( \Pi_{11} \in \mathbb{C}^{n \times n} \). By the GSVD of \{\mathbf{A}, \mathbf{B}\} in (1.1) and (1.2) we have

\[
\max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 \mathbf{A}(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \Pi_2 \mathcal{G}_i)|

= \max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 U \Sigma_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i)|

= \max_{\Pi_{11} \Pi_{12} \leq I_n, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_{11} \hat{\Sigma}_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i)| .
\]

Observe \( f(\Pi_{11}) = \text{tr}(\Pi_{11} \hat{\Sigma}_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i) \) is an analytical function of several complex variables on the domain \( \Pi_{11} \Pi_{11}^H \leq I_n \). By Lemma 2.2 we know that \( f(\Pi_{11}) \) attains its maximum modulus on the characteristic manifold \( \{\Pi_{11} \in \mathbb{C}^{n \times n} : \Pi_{11} \Pi_{11}^H = I_n\} \). Therefore,

\[
\max_{\Pi_{11} \Pi_{11} \leq I_n, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_{11} \hat{\Sigma}_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i)|

= \max_{\Pi_{11} \Pi_{11}^H = I_n, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_{11} \hat{\Sigma}_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i)|

\]

and it follows from Lemma 2.1 that

\[
\max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 \mathbf{A}(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \Pi_2 \mathcal{G}_i)|

= \max_{\Pi_{11} \Pi_{11}^H = I_n, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_{11} \hat{\Sigma}_A R(R^H R)^{-1/2} \Pi_2 \mathcal{G}_i)|

= \alpha_1 + \cdots + \alpha_i.
\]

(2.25)

Similarly, we have

\[
\max_{\Pi_1 \in \mathbb{U}_m, \Pi_2 \in \mathbb{U}_n} |\text{tr}(\Pi_1 \mathbf{A}(\mathbf{A}^H \mathbf{A} + \mathbf{B}^H \mathbf{B})^{-1/2} \Pi_2 \mathcal{G}_{i-1})| = \alpha_1 + \cdots + \alpha_{i-1} .
\]

(2.26)
From (2.25) and (2.26) we know that (2.17) holds for 1 \leq i \leq n.

b) If n \leq p, then for 1 \leq i \leq n, let \mathcal{H}_i be defined by (2.20). We set \Sigma_B = \begin{pmatrix} O_{(p-n)\times n} \end{pmatrix}. For any \Xi_1 \in U_p, let \Xi_1V = \begin{pmatrix} \Xi_{11} & \Xi_{12} \\ \Xi_{21} & \Xi_{22} \end{pmatrix} \in U_p with \Xi_{11} \in \mathbb{C}^{n \times n}. By the GSVD of \{A, B\} in (1.1) and (1.2), Lemma 2.2 and Lemma 2.1 we have

\[
\max_{\Xi_1 \in U_p, \Xi_2 \in U_n} |\text{tr}(\Xi_1B(A^HA + B^HB)^{-1/2}\Xi_2\mathcal{H}_i)| - \max_{\Xi_1 \in U_p, \Xi_2 \in U_n} |\text{tr}(\Xi_1V\Sigma_BR(R^HR)^{-1/2}\Xi_2\mathcal{H}_i)| = \max_{\Xi_{22} \leq \mathcal{I}_n, \Xi_2 \in U_n} |\text{tr}(\Xi_{22}\hat{\Sigma}_BR(R^HR)^{-1/2}\Xi_2\mathcal{H}_i)| - \max_{\Xi_{22} = \mathcal{I}_n, \Xi_2 \in U_n} |\text{tr}(\Xi_{22}\hat{\Sigma}_BR(R^HR)^{-1/2}\Xi_2\mathcal{H}_i)| = \beta_1 + \cdots + \beta_n. \tag{2.27}
\]

Similarly, we have

\[
\max_{\Xi_1 \in U_p, \Xi_2 \in U_n} |\text{tr}(\Xi_1B(A^HA + B^HB)^{-1/2}\Xi_2\mathcal{H}_{i+1})| = \beta_{i+1} + \cdots + \beta_n. \tag{2.28}
\]

Using (2.27) and (2.28) we know that (2.19) holds for 1 \leq i \leq n.

c) If m \leq n, then it is easy to check that \alpha_{m+1} = \cdots = \alpha_n = 0. For 1 \leq i \leq m, let \mathcal{S}_i be defined by (2.22). We set \Sigma_A = (\hat{\Sigma}_A, O_{m \times (n-m)}) and \Pi_3(R^HR)^{-1/2}R^H = \begin{pmatrix} \hat{\Pi}_{11} & \hat{\Pi}_{12} \\ \hat{\Pi}_{21} & \hat{\Pi}_{22} \end{pmatrix} \in U_n.

By the GSVD of \{A, B\} in (1.1) and (1.2), Lemma 2.2 and Lemma 2.1 we have

\[
\max_{\Pi_3 \in U_m, \Pi_4 \in U_m} |\text{tr}(\Pi_3(A^HA + B^HB)^{-1/2}A^H\Pi_4\mathcal{S}_i)| = \max_{\Pi_3 \in U_m, \Pi_4 \in U_m} |\text{tr}(\Pi_3(R^HR)^{-1/2}R^H\Sigma^H_3U^H\Pi_4\mathcal{S}_i)| = \max_{\hat{\Pi}_{11} \leq \mathcal{I}_m, \Pi_4 \in U_m} |\text{tr}(\hat{\Pi}_{11}\hat{\Sigma}^H_AU^H\Pi_4\mathcal{S}_i)| = \max_{\hat{\Pi}_{11} = \mathcal{I}_m, \Pi_4 \in U_m} |\text{tr}(\hat{\Pi}_{11}\hat{\Sigma}^H_AU^H\Pi_4\mathcal{S}_i)| = \alpha_1 + \cdots + \alpha_i. \tag{2.29}
\]

Similarly, we have

\[
\max_{\Pi_3 \in U_m, \Pi_4 \in U_m} |\text{tr}(\Pi_3(A^HA + B^HB)^{-1/2}A^H\Pi_4\mathcal{S}_{i-1})| = \alpha_1 + \cdots + \alpha_{i-1}. \tag{2.30}
\]

From (2.29) and (2.30) we know that (2.21) holds for 1 \leq i \leq m.

d) If p \leq n, then we know that \beta_1 = \cdots = \beta_{n-p} = 0. For n - p + 1 \leq i \leq n, let \mathcal{W}_i be defined by (2.24). We set \Sigma_B = (O_{p \times (n-p)}, \hat{\Sigma}_B) and \Xi_3(R^HR)^{-1/2}R^H = \begin{pmatrix} \tilde{\Xi}_{11} & \tilde{\Xi}_{12} \\ \tilde{\Xi}_{21} & \tilde{\Xi}_{22} \end{pmatrix} \in U_n. By
Similarly, we have \( \text{the GSVD of } \{A, B\} \) in (1.1) and (1.2). Lemma 2.2 and Lemma 2.1 we have
\[
\max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3(A^H A + B^H B)^{-1/2}B^H \Xi_4 W_i)| = \max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3(R^H R)^{-1/2}R^H \Sigma^H B^H \Xi_4 W_i)|
\]
\[
= \max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3 \hat{H}_B^H \Sigma^H B^H \Xi_4 W_i)| = \max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3 \hat{H}_B^H \Sigma^H B^H \Xi_4 W_i)|
\]
\[
= \max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3 \hat{H}_B^H \Sigma^H B^H \Xi_4 W_i)| = \beta_i + \cdots + \beta_n. \tag{2.31}
\]

Similarly, we have
\[
\max_{\Xi \in U_n, \Xi_4 \in U_p} |\text{tr}(\Xi_3(A^H A + B^H B)^{-1/2}B^H \Xi_4 W_{i+1})| = \beta_{i+1} + \cdots + \beta_n. \tag{2.32}
\]

Using (2.31) and (2.32) we can conclude that (2.23) holds for \( n - p + 1 \leq i \leq n \). The proof is complete.

Based on Lemmas 2.3 and 2.4, we give new model formulations of any GSV \((\alpha_i, \beta_i)\) of the \((m, p, n)\)-GMP \(\{A, B\}\). We note that \( \alpha_i^2 + \beta_i^2 = 1 \) for \( 1 \leq i \leq n \) and for \( m \leq n \), \( \alpha_{m+1} = \cdots = \alpha_n = 0 \) and for \( p \leq n \), \( \beta_1 = \cdots = \alpha_{n-p} = 0 \). Then, for \( 1 \leq i \leq [n/2] \) we use \( \alpha_i \)'s formulations and for \([n/2]+1 \leq i \leq n \) we use \( \beta_i \)'s formulations. Therefore, we can derive the following results.

**Theorem 2.5** Let \( \{A, B\} \) be an \((m, p, n)\)-GMP and the GSV \((\alpha_i, \beta_i)\) of \(\{A, B\}\) be given in (1.1) and (1.2). The following conclusions hold true.

a) If \( n \leq m \), then we use
\[
\alpha_i = \phi_i^\frac{1}{2} (\tilde{\phi}_i), \text{ for } 1 \leq i \leq [n/2].
\]

b) If \( m < n \) and further if \( m < [n/2] \), then we use
\[
\alpha_i = \phi_i^\frac{1}{2} (\tilde{\phi}_i), \text{ for } 1 \leq i \leq m, \text{ } \alpha_{m+1} = \cdots = \alpha_{[n/2]} = 0;
\]

If \( m < n \) and further if \([n/2] \leq m \), then we use
\[
\alpha_i = \phi_i^\frac{1}{2} (\tilde{\phi}_i), \text{ for } 1 \leq i \leq [n/2].
\]

c) If \( n \leq p \), then we use
\[
\beta_i = \psi_i^\frac{1}{2} (\tilde{\psi}_i), \text{ for } [n/2] + 1 \leq i \leq n.
\]

d) If \( p < n \) and further if \( n - p < [n/2] \), then we use
\[
\beta_i = \chi_i^\frac{1}{2} (\tilde{\chi}_i), \text{ for } [n/2] + 1 \leq i \leq n;
\]

If \( p < n \) and further if \( n - p > [n/2] \), then we use
\[
\beta_{[n/2]+1} = \cdots = \beta_{n-p} = 0, \text{ } \beta_i = \chi_i^\frac{1}{2} (\tilde{\chi}_i), \text{ for } n - p + 1 \leq i \leq n.
\]
Remark 2.6 By Lemma 2.3, we can provide new formula model of any GSV \((\alpha_i, \beta_i)\) of the GMP \(\{A, B\}\) by trace function under one variable. By Lemma 2.4, we can provide new formula model of any GSV \((\alpha_i, \beta_i)\) of the GMP \(\{A, B\}\) by trace function under two variables. By Theorem 2.5 and Lemma 2.3, we know that for \(1 \leq i \leq \lfloor n/2 \rfloor\), we use \(\alpha_i\)'s formula referring to \(Q_i, Q_i\) involving more zeros (about \(n - i\) or \(m - i\) zeros) as their diagonal entries and for \(\lfloor n/2 \rfloor + 1 \leq i \leq n\), we use \(\beta_i\)'s formula referring to \(P_i, P_i\) involving more zeros (about \(i\) zeros) as their diagonal entries by trace function under one variable. By Theorem 2.5 and Lemma 2.4, we know that for \(1 \leq i \leq \lfloor n/2 \rfloor\), we use \(\alpha_i\)'s formula referring to \(G_i, S_i\) involving more zeros (about \(n - i, m - i\) zeros) as their diagonal entries by trace function under one variable.

3 Numerical methods for computing \(\alpha_i\) and \(\beta_i\)

In this section, based on trace function optimization under one variable and two variables, we give computational methods for computing \(\alpha_i\) and \(\beta_i\), respectively.

3.1 Based on trace function optimization under one variable

In this subsection, based on trace function optimization under one variable, we present Newton's method on Grassmann manifold for computing \(\alpha_i\) and \(\beta_i\) by Theorem 2.5 and Lemma 2.3.

Let \(\{A, B\}\) be an \((m, p, n)\)-GMP. For demonstration purpose, we assume that \(n \leq m\) and \(n \leq p\). To compute \(\alpha_i^2\) as defined in Theorem 2.5, we consider the following matrix optimization problems:

\[
\begin{align*}
\max & \quad f_1(\Phi_1) := \frac{1}{2} \text{tr}(A^H \Phi_1^HQ_1 \Phi_1 A (A^H A + B^H B)^{-1}) \\
\text{s.t.} & \quad \Phi_1 \in \mathbb{U}_m
\end{align*}
\]

and

\[
\begin{align*}
\max & \quad f_2(\Phi_2) := \frac{1}{2} \text{tr}(B^H \Phi_2^HP_1 \Phi_2 B (A^H A + B^H B)^{-1}) \\
\text{s.t.} & \quad \Phi_2 \in \mathbb{U}_p
\end{align*}
\]

By using the definitions of \(Q_i\) and \(P_i\) and \(\alpha_i^2 + \beta_i^2 = 1\) for \(1 \leq i \leq n\), we only need to focus on the following matrix optimization problems:

\[
\begin{align*}
\max & \quad f_1(\Phi_{11}) := \frac{1}{2} \text{tr}(\Phi_{11}^H C \Phi_{11}) \\
\text{s.t.} & \quad \Phi_{11} \in \text{St}(m, i) := \{ \Phi_{11} \in \mathbb{C}^{m \times i} \mid \Phi_{11}^H \Phi_{11} = I_i \},
\end{align*}
\]

and

\[
\begin{align*}
\max & \quad f_2(\Phi_{22}) := \frac{1}{2} \text{tr}(\Phi_{22}^H D \Phi_{22}) \\
\text{s.t.} & \quad \Phi_{22} \in \text{St}(p, n - i + 1),
\end{align*}
\]

for \(1 \leq i \leq \lfloor n/2 \rfloor\), where \(C = A(A^H A + B^H B)^{-1})A^H\), \(D = B(A^H A + B^H B)^{-1})B^H\), \(\Phi_{11}\) is the first \(i\) columns of \(\Phi_1^H\) and \(\Phi_{22}\) is the last \(n - i + 1\) columns of \(\Phi_2^H\).
In the following, we focus on the solution of the matrix optimization problem (3.3). The matrix optimization problem (3.4) can be solved in a similar way.

We note that \( f_1(\Phi_{11}Q) = f_1(\Phi_{11}) \) for all \( \Phi_{11} \in \text{St}(m,i) \) and \( Q \in \mathbb{U}_i \). As in [16], let the complex Grassmann manifold \( \text{Grass}(m,i) \) be the set of all \( i \)-dimensional complex subspace of \( \mathbb{C}^m \). If \([X]\) means the subspace spanned by the columns of \( X \in \text{St}(m,i) \), then we have \([X] \in \text{Grass}(m,i)\). In particular, for any \( X \in \text{St}(m,i) \), the natural projection \([X] \in \text{Grass}(m,i)\) corresponds the equivalent class \( \{XQ \in \text{St}(m,i) \mid Q \in \mathbb{U}_i\} \) of \( \text{St}(m,i) \). Thus, instead of problem (3.3), we consider the following optimization problem:

\[
\begin{align*}
\text{max} & \quad f_1([\Phi_{11}]) := f_1(\Phi_{11}) \\
\text{s.t.} & \quad [\Phi_{11}] \in \text{Grass}(m,i).
\end{align*}
\]

(3.5)

Next, we present Newton’s method for solving the optimization problem (3.5). Let \( \Phi_{11} \in \text{St}(m,i) \). The tangent space of \( \text{St}(m,i) \) at \( \Phi_{11} \) is given by [16]

\[
\mathcal{T}_{\Phi_{11}}\text{St}(m,i) = \{Z \in \mathbb{C}^{m \times i} \mid Z = \Phi_{11}\Omega + (\Phi_{11})_{\bot}K, \Omega^H = -\Omega, \Omega \in \mathbb{C}^{i,i}, K \in \mathbb{C}^{m-i,i}\},
\]

which can be endowed with the inner product

\[
\langle Z_1, Z_2 \rangle = \Re[\text{tr}(Z_2^H(I - \frac{1}{2}\Phi_{11}\Phi_{11}^H)Z_1)], \quad Z_1, Z_2 \in \mathcal{T}_{\Phi_{11}}\text{St}(m,i), \Phi_{11} \in \text{St}(m,i).
\]

Here, \((\Phi_{11})_{\bot}\) means that \(\text{span}((\Phi_{11})_{\bot})\) is the orthogonal complement of \(\text{span}(\Phi_{11})\), where \(\text{span}(\Phi_{11})\) denotes a linear space spanned by the column vectors of \(\Phi_{11}\). We note that the tangent space of \(\text{Grass}(m,i)\) at \([\Phi_{11}]\) is given by [16]

\[
\mathcal{T}_{[\Phi_{11}]}\text{Grass}(m,i) = \{Z \in \mathbb{C}^{m \times i} \mid Z = (\Phi_{11})_{\bot}K, K \in \mathbb{C}^{m-i,i}\} \subset \mathcal{T}_{\Phi_{11}}\text{St}(m,i).
\]

Hence, we can define a Riemannian metric on \(\text{Grass}(m,i)\) by

\[
\langle Z_1, Z_2 \rangle = \Re[\text{tr}(Z_2^HZ_1)], \quad Z_1, Z_2 \in \mathcal{T}_{[\Phi_{11}]}\text{Grass}(m,i), \Phi_{11} \in \text{St}(m,i)
\]

with the induced norm \(\|\cdot\|\). Then the orthogonal projection onto \(\mathcal{T}_{[\Phi_{11}]}\text{Grass}(m,i)\) is given by

\[
P_{\Phi_{11}}Z = (I - \Phi_{11}\Phi_{11}^H)Z, \quad \forall Z \in \mathbb{C}^{m \times i}.
\]

We define the local cost function \(g : \mathcal{T}_{[\Phi_{11}]}\text{Grass}(m,i) \to \mathbb{R}\) by

\[
g(Z) = \tilde{f}_1([\Phi_{11} + Z]).
\]

It is easy to check that, for any \(Z \in \mathcal{T}_{[\Phi_{11}]}\text{Grass}(m,i)\),

\[
g(Z) = f_1(\Phi_{11}) + \Re[\text{tr}(Z^HD_{\Phi_{11}}) + \frac{1}{2}\text{vec}(Z)^H(H_{\Phi_{11}} - (\Phi_{11}^HD_{\Phi_{11}})^T \otimes I_m)\text{vec}(Z)),
\]

where \(D_{\Phi_{11}} = C\Phi_{11}\) and \(H_{\Phi_{11}} = I_i \otimes C\) are the derivative and Hessian of \(f_1\) at \(\Phi_{11}\), respectively.

Based on the above analysis, Newton’s method for solving the optimization problem (3.5) can be described as follows [16].
Algorithm 3.1 Step 0. Choose $\Phi_{11}^0 \in \text{St}(m,i)$, $\rho, \eta \in (0,1)$, $\sigma \in (0,1/2]$ and let $k := 1$.

Step 1. Apply the conjugate gradient (CG) method [10, Chap. 11.3] to solving

$$P_{\Phi_{11}^k}(CZ^k - Z^k(\Phi_{11}^k)^HD_{\Phi_{11}^k}) = -P_{\Phi_{11}^k}D_{\Phi_{11}^k}$$

for $Z^k \in T_{\Phi_{11}^k}\text{Grass}(m,i)$ such that

$$\|P_{\Phi_{11}^k}(CZ^k - Z^k(\Phi_{11}^k)^HD_{\Phi_{11}^k}) + P_{\Phi_{11}^k}D_{\Phi_{11}^k}\| \leq \eta\|P_{\Phi_{11}^k}D_{\Phi_{11}^k}\| \quad (3.6)$$

and

$$\langle P_{\Phi_{11}^k}D_{\Phi_{11}^k}, Z^k \rangle \leq \eta_k\langle Z^k, Z^k \rangle \quad (3.7)$$

where $\eta_k = \min\{\eta, \|P_{\Phi_{11}^k}D_{\Phi_{11}^k}\|\}$. If (3.6) and (3.7) are not attainable, then let

$$Z^k = -P_{\Phi_{11}^k}D_{\Phi_{11}^k}.$$

Step 2. Let $l_k > 0$ be the smallest integer $m$ such that

$$f_1(\pi(\Phi_{11}^k + \beta^l Z^k)) \leq f_1(\Phi_{11}^k) + \sigma \rho^l \langle P_{\Phi_{11}^k}D_{\Phi_{11}^k}, Z^k \rangle$$

Set

$$Z^{k+1} = \pi(\Phi_{11}^k + \beta^l Z^k).$$

Step 3. Replace $k$ by $k + 1$ and go to Step 1.

In Step 2 of Algorithm 3.1, $\pi : \mathbb{C}^{m \times i} \to \text{Grass}(m,i)$ is the projection onto Grass$(m,i)$, which is defined as follows: Let $X \in \mathbb{C}^{m \times i}$ be full column rank. Then

$$\pi(X) = \left[\text{argmin}_{Y \in \text{St}(m,i)} \|X - Y\|^2\right].$$

As noted in [16], if the SVD of $X$ is given by $X = U\Sigma V^H$, then $\pi(X) = UI_{m,i}V^H$. If the QR decomposition of $X$ is $X = QR$, then $\pi(X) = Q_{m,i}$.

We must point out that, if $A$ and $B$ are real matrices, then the matrix $C$ in problem (3.3) is real. In this case, one may use the trust-region methods in [1] or the geometric Newton algorithm in [27] for solving the optimization problem (3.3).

3.2 Based on trace function optimization under two variables

In the following, based on trace function optimization under two variables, we use classical Golub-Kahan bidiagonalization method (which is implemented by MATLAB-routine function svds) for computing $\alpha_i$ and $\beta_i$ by Theorem 2.5 and Lemma 2.4. For simplicity, we assume that $n \leq m$ and $n \leq p$. To compute $\alpha_i$ and $\beta_i$ as defined in Lemma 2.4, we study the following matrix optimization problems:

$$\max g_1(\Pi_1, \Pi_2) := |\text{tr}(\Pi_1 A(A^HA + B^HB)^{-1/2}\Pi_2 G_i)|$$

s.t. $\Pi_1 \in U_m, \Pi_2 \in U_n$ \quad (3.8)
and

\[
\max g_2(\Xi_1, \Xi_2) := |\text{tr}(\Xi_1 B (A^H A + B^H B)^{-1/2} \Xi_2 H)|
\]

s.t. \( \Xi_1 \in U_p, \Xi_2 \in U_n \).

By using the definitions of \( G_i \) and \( H_i \) and \( \alpha_i^2 + \beta_i^2 = 1 \) for \( 1 \leq i \leq n \), we only need to focus on the following matrix optimization problems:

\[
\max g_1(\Pi_{11}, \Pi_{21}) := |\text{tr}(\Pi_{11}^H E \Pi_{21})|
\]

s.t. \( \Pi_{11} \in \text{St}(m, i), \Pi_{21} \in \text{St}(n, i) \) \hspace{1cm} (3.10)

and

\[
\max g_2(\Xi_{12}, \Xi_{22}) := |\text{tr}(\Xi_{12}^H F \Xi_{22})|
\]

s.t. \( \Phi_{22} \in \text{St}(p, n - i + 1) \) \hspace{1cm} (3.11)

for \( 1 \leq i \leq \lfloor n/2 \rfloor \), where \( E = A(A^H A + B^H B)^{-1/2} \), \( F = B(A^H A + B^H B)^{-1/2} \), \( \Pi_{11}^H \) is the first \( i \) rows of \( \Phi_1 \), \( \Pi_{21} \) is the first \( i \) columns of \( \Pi_2 \), \( \Xi_{12}^H \) is the last \( n - i + 1 \) rows of \( \Xi_1 \), and \( \Xi_{22} \) is the last \( n - i + 1 \) columns of \( \Xi_2 \). We will discuss the solution of the matrix optimization problem (3.10). The matrix optimization problem (3.11) can be solved similarly.

As in [10, Chap.10.4], starting a unit 2-norm vector \( v_1 \in \mathbb{C}^n \), the Golub-Kahan process [9] for bidiagonalizing the matrix \( E \) can be stated as follows (see also [10, Algorithm 10.4.1]).

**Algorithm 3.2**

**Step 0.** Choose \( v_1 \in \mathbb{C}^n \) with \( \|v_1\|_2 = 1 \), \( p_0 = v_1 \), \( \beta_0 = 1 \), \( u_0 = 0 \), and let \( k := 0 \).

**Step 1.** While \( \beta_k \neq 0 \)

\[
\begin{align*}
v_{k+1} &= p_k/\beta_k \\
r_{k+1} &= Av_{k+1} - \beta_k u_k \\
\alpha_{k+1} &= \|r_{k+1}\|_2 \\
u_{k+1} &= r_{k+1}/\alpha_{k+1} \\
p_{k+1} &= A^H u_{k+1} - \alpha_{k+1} v_{k+1} \\
\beta_{k+1} &= \|p_{k+1}\|_2
\end{align*}
\]

**end**

**Step 2.** Replace \( k \) by \( k + 1 \) and go to Step 1.

From Algorithm 3.2 we can obtain \( V_k = [v_1, v_2, \ldots, v_k] \), \( U_k = [u_1, u_2, \ldots, u_k] \) with orthonormal columns and a \( k \times k \) bidiagonal matrix

\[
B_k = \begin{bmatrix}
\alpha_1 & \beta_1 & \cdots & 0 \\
0 & \alpha_2 & \beta_2 & \cdots \\
\vdots & \ddots & \ddots & \ddots \\
0 & \cdots & 0 & \alpha_{k-1} & \beta_{k-1} \\
0 & \cdots & 0 & 0 & \alpha_k
\end{bmatrix}
\]
such that
\[ EV_k = U_k B_k \quad \text{and} \quad E^H U_k = V_k B^H + p_k e_k^T. \]
(3.12)

Then we can compute the SVD of \( B_k \) via the SVD algorithm in \([10, \text{Chap. 8.6}]\)
\[ F_k^H B_k G_k = \Gamma_k = \text{diag}(\gamma_1, \ldots, \gamma_k) \]
and form the matrices
\[ Y_k = V_k G_k \quad \text{and} \quad Z_k = U_k F_k. \]

From (3.12) we have
\[ EY_k = Z_k \Gamma_k \quad \text{and} \quad E^H Z_k = Y_k \Gamma + p_k e_k^T F_k. \]
Hence, the singular vector matrix \( \Pi_{11} \) and \( \Pi_{21} \) can be estimated from the \( i \) column vectors of \( Z_k \) and \( Y_k \) corresponding to the \( i \) largest singular values of \( B_k \). We then use \textsc{Matlab}-routine function \textsc{svds}. Commands `tt \textsc{svds}(A,i)` and `\textsc{svds}(A,i,’smallest’)’ denote \( i \) largest and smallest singular values of matrix \( A \), respectively.

**Remark 3.3** The given model formulas by trace function optimization under one variable and two variables are new. Based on the new formulas, we give computational methods involving Newton’s method on Grassmann manifold and Algorithm 10.4.1 in \([10]\). Based on our new model formulas, the proposed algorithms are very efficient for computing arbitrary GSV of an \((m,p,n)-\text{GMP} \{A,B\}\) for general scaled GSV problems: (i) For large \( n \), we need design more effective algorithms to compute arbitrary GSV of the GMP for large scaled GSV problems. This is a new topic in the future. (ii) For \( A^H A + B^H B \) being ill-posed (i.e., \( \sigma_{\max}(A^H A + B^H B) \) is far away from \( \sigma_{\min}(A^H A + B^H B) \)), we need construct more effective algorithms. This is also a new topic in the future. (iii) The convergent analysis and theoretical computational complexity of the proposed algorithms are also discussed as a new topic in the future.

4 Numerical experiments

In this section we give numerical experiments to illustrate the efficiency of Algorithms 3.1 and 3.2 for computing arbitrary GSV of a GMP. All the tests were carried out in \textsc{Matlab} 2019a running on a workstation with a Intel Xeon CPU E5-2687W at 3.10 GHz and 32 GB of RAM. In particular, Algorithm 3.2 was implemented by the built-in functions \textsc{svds} in \textsc{Matlab} R2019a.

In our numerical tests, the tolerance for \textsc{svds} is set to be \( 10^{-12} \) while the stopping criterion for Algorithm 3.1 is set to be
\[ \| P_{\Psi_{11}} D_{\Psi_{11}} \| \leq 10^{-6} \]
and we set \( \eta = 0.1, \rho = 0.5, \) and \( \sigma = 10^{-4} \). The largest number of outer iterations in Algorithm 3.1 is set to be 100 and the largest number of iterations in the CG method is set to be \( mn \) or \( np \). We will assess efficiency of the proposed algorithms.

We first give some random examples to assess the efficiency of the proposed algorithms.

**Example 4.1** We first generate the exact generalized singular values \( \alpha_1^*, \ldots, \alpha_n^* \) and \( \beta_1^*, \ldots, \beta_n^* \) by using the built-in function `\texttt{rand}` in \textsc{Matlab} R2019a such that
\[ 1 \geq \alpha_1^* \geq \alpha_2^* \geq \cdots \geq \alpha_n^* \geq 0, \quad 0 \leq \beta_1^* \leq \beta_2^* \leq \cdots \leq \beta_n^* \leq 1, \]
and 
\[ \alpha_j^2 + \beta_j^2 = 1, \quad 1 \leq j \leq n. \]

Then the \((m, p, n)\)-GMP \(\{A, B\}\) is given by \(A = U_\ast \Sigma_A \ast W_\ast\) and \(B = V_\ast \Sigma_B \ast W_\ast\), where
\[
\Sigma_A = \begin{pmatrix} \text{diag}(\alpha_1, \ldots, \alpha_n) \\ O_{(m-n) \times n} \end{pmatrix}, \quad \Sigma_B = \begin{pmatrix} O_{(p-n) \times n} \\ \text{diag}(\beta_1^*, \ldots, \beta_n^*) \end{pmatrix},
\]
and the orthogonal matrices \(U_\ast \in \mathbb{C}^{m \times m}\) and \(V_\ast \in \mathbb{C}^{p \times p}\) and the nonsingular matrix \(W_\ast \in \mathbb{R}^{n \times n}\) are generated by the built-in functions \texttt{orth} and \texttt{randn} in MATLAB R2019a.

For demonstration purposes, we assume that \(n \leq m\) and \(n \leq p\). Table 4.1 lists the numerical results for Example 4.1, where ‘Err1.’, ‘Err2.’, ‘Err3.’, ‘Err4.’, and ‘CT.’ mean the relative errors \(\max_{1 \leq k \leq \lfloor n/2 \rfloor} |\alpha_k - \alpha_k^*|/|\alpha_k^*|\), \(\max_{\lfloor n/2 \rfloor + 1 \leq i \leq n} |\beta_k - \beta_k^*|/|\beta_k^*|\), \(\min_{1 \leq i \leq \lfloor n/2 \rfloor} |\alpha_k - \alpha_k^*|/|\alpha_k^*|\), \(\min_{\lfloor n/2 \rfloor + 1 \leq i \leq n} |\beta_k - \beta_k^*|/|\beta_k^*|\), and the total computing time in seconds at the final iterates of the corresponding algorithms accordingly.

| Alg. | \((m, p, n)\) | \text{Err1.} | \text{Err2.} | \text{Err3.} | \text{Err4.} | CT. |
|------|--------|--------|--------|--------|--------|-----|
| 3.1  | (100, 80, 20) | 1.02 \times 10^{-14} | 2.15 \times 10^{-14} | 1.79 \times 10^{-16} | 1.74 \times 10^{-17} | 14.62 |
|      | (200, 180, 40) | 4.90 \times 10^{-14} | 2.16 \times 10^{-13} | 2.63 \times 10^{-16} | 2.70 \times 10^{-16} | 19.80 |
|      | (500, 400, 80) | 5.28 \times 10^{-14} | 3.52 \times 10^{-13} | 5.58 \times 10^{-17} | 2.58 \times 10^{-16} | 20.35 |
|      | (800, 700, 100) | 4.38 \times 10^{-14} | 2.53 \times 10^{-14} | 2.75 \times 10^{-17} | 3.85 \times 10^{-17} | 21.18 |
|      | (1000, 800, 300) | 5.06 \times 10^{-14} | 3.72 \times 10^{-14} | 2.24 \times 10^{-16} | 1.09 \times 10^{-17} | 23.78 |
|      | (2000, 1800, 500) | 2.58 \times 10^{-14} | 3.47 \times 10^{-14} | 2.35 \times 10^{-16} | 3.06 \times 10^{-16} | 30.05 |
|      | (3000, 2500, 600) | 3.20 \times 10^{-13} | 3.11 \times 10^{-14} | 1.83 \times 10^{-15} | 2.96 \times 10^{-16} | 38.10 |
|      | (5000, 4000, 600) | 1.38 \times 10^{-12} | 2.16 \times 10^{-12} | 4.70 \times 10^{-15} | 2.63 \times 10^{-14} | 42.07 |
|      | (10000, 8000, 600) | 1.84 \times 10^{-13} | 5.26 \times 10^{-14} | 3.28 \times 10^{-15} | 1.58 \times 10^{-16} | 45.59 |

| svds | (100, 80, 20) | 4.62 \times 10^{-12} | 1.47 \times 10^{-14} | 2.53 \times 10^{-16} | 3.89 \times 10^{-17} | 15.10 |
|      | (200, 180, 40) | 6.28 \times 10^{-14} | 7.90 \times 10^{-13} | 3.37 \times 10^{-16} | 4.07 \times 10^{-16} | 20.03 |
|      | (500, 400, 80) | 6.83 \times 10^{-14} | 6.02 \times 10^{-13} | 6.84 \times 10^{-17} | 6.94 \times 10^{-15} | 21.75 |
|      | (800, 700, 100) | 5.40 \times 10^{-14} | 3.11 \times 10^{-14} | 3.01 \times 10^{-17} | 5.57 \times 10^{-17} | 22.94 |
|      | (1000, 800, 300) | 7.55 \times 10^{-14} | 5.09 \times 10^{-14} | 2.36 \times 10^{-16} | 8.46 \times 10^{-16} | 24.80 |
|      | (2000, 1800, 500) | 4.05 \times 10^{-14} | 5.38 \times 10^{-14} | 2.96 \times 10^{-16} | 5.88 \times 10^{-16} | 24.52 |
|      | (3000, 2500, 600) | 4.67 \times 10^{-13} | 5.93 \times 10^{-14} | 2.74 \times 10^{-15} | 4.27 \times 10^{-16} | 25.09 |
|      | (5000, 4000, 600) | 4.52 \times 10^{-12} | 3.85 \times 10^{-12} | 2.07 \times 10^{-14} | 1.44 \times 10^{-13} | 25.95 |
|      | (10000, 8000, 600) | 3.08 \times 10^{-13} | 1.05 \times 10^{-13} | 6.93 \times 10^{-15} | 5.96 \times 10^{-16} | 26.03 |

Also, in Figures 4.1, 4.3, we report the corresponding values of ‘Err1.’, ‘Err2.’, ‘Err3.’, ‘Err4.’, and ‘CT.’ for Example 4.1 with other random \((m, p, n)\)-GMPs.

We see from Table 4.1 that the computational time by Algorithm 3.1 is less than 45.59s and computational time by \texttt{svds} is less than 26.03s. From Table 4.1 and Figures 4.1, 4.2 we observe that all values of \(\max_{1 \leq i \leq \lfloor n/2 \rfloor} |\alpha_k - \alpha_k^*|/|\alpha_k^*|\), \(\max_{\lfloor n/2 \rfloor + 1 \leq i \leq n} |\beta_k - \beta_k^*|/|\beta_k^*|\) and \(\min_{1 \leq i \leq \lfloor n/2 \rfloor} |\alpha_k - \alpha_k^*|/|\alpha_k^*|\), \(\min_{\lfloor n/2 \rfloor + 1 \leq i \leq n} |\beta_k - \beta_k^*|/|\beta_k^*|\) computed by Algorithm 3.1 are sharper than those computed by \texttt{svds} for most cases. This implies the solution precision of Algorithm 3.1 is a little higher than the solver \texttt{svds}. From Figure 4.3 we see that the computational time of \texttt{svds} is a little less than Algorithm 3.1 for some random GMPs.

We then give some practical numerical examples to access the efficiency of the proposed model and algorithms.
Example 4.2 We derive data of the corresponding gene mRNA expression level after doing the mice macrophage research experiment. Several groups of datasets from different conditions with different chemical reagents, which are denoted by \{E_1, F_1\} and \{E_2, F_2\} as in Appendix.

As we know in [2, 3], the GSVD induces a linear transformation of two data sets from the two genes \( \times \) arrays spaces to two reduced and diagonalized genelets \( \times \) arraylets spaces. The genelets are shared by both data sets. A single microarray probes the relative expression levels of \( m \) genes in a single sample. A series of \( n_1 \) arrays probes the genome-scale expression levels in \( n_1 \) different samples, i.e., under \( n_1 \) different experimental conditions. Let the matrix \( A \), of size \( m \)-genes \( \times \) \( n_1 \)-arrays, denote the full expression data, whose the \( k \)-th row is the expression of the \( k \)-th gene across the different samples that correspond to the different arrays.

Let the matrix \( B \), of size \( p \)-genes \( \times \) \( n_2 \)-arrays, denote the relative expression levels of \( p \) genes under \( n_2 = n_1 = n < \max\{m, p\} \) experimental conditions that correspond one to one to the \( n_1 \) conditions underlying \( A \). The GSVD then induces simultaneous linear transformation of
two expression data sets $A$ and $B$ from two $m$-genes × $n$-arrays and $p$-genes × $n$-arrays spaces to two reduced $n$-genelets × $n$-arraylets spaces. For more details, please see the PNAS web site: www.pnas.org, and also http://genome-www.stanford.edu/GSVD/). Let $A$ and $B$ have the following GSVD:

$$A = U \Sigma_A R^{-1} \quad \text{and} \quad B = V \Sigma_B R^{-1}.$$

In these spaces the data is represented by the diagonal non-negative matrices $\Sigma_A$ and $\Sigma_B$ and denote $\langle k|\Sigma_A|l \rangle = \varepsilon_{1,l} \delta_{kl}$, $\langle k|\Sigma_B|l \rangle = \varepsilon_{2,l} \delta_{kl}$ for all $1 \leq k, l \leq n$, where $\delta_{kl} = 1$ if $k = l$ and $\delta_{kl} = 0$ if $k \neq l$. By $\langle k|\Sigma_A$ we denote the $k$-th row of the matrix $\Sigma_A$, which lists the expression of the $k$-th gene across the different samples that correspond to the different arrays. By $\Sigma_A|l \rangle$ we denote the $l$-th column of the matrix $\Sigma_A$, which lists the genome-scale expression measured by the $l$-th array. The antisymmetric angular distance between the data sets,

$$\theta_l = \arctan(\varepsilon_{1,l}/\varepsilon_{2,l}) - \pi/4,$$

indicates the relative significance of the $n$th genelet, i.e., its significance in the first data set relative to that in the second data set in terms of the ratio of the expression information captured by this genelet in the first data set to that in the second data set. An angular distance of 0 indicates a genelet of equal significance in both data sets, with $\pm \pi/4$ ($\pm 0.785398163397448$) indicates no significance in the second data set relative to the first, or in the first relative to the second, respectively. (For more details, please see the PNAS web site).

In this experiment expression levels of 10020 genes are probed in a single sample. A series of $n$ arrays probes the genome-scale expression levels in $n$ samples under different conditions. By $E_1, F_1, E_2, F_2$ we denote gene mRNA expression data with original data set, with just PA stimulation and after knocking out SHP2 gene of mice without and with PA stimulation, respectively. Data sets $\{E_1, F_1\}$ and $\{E_2, F_2\}$ with $m = 10020$, $p = 10020$ and $n = 128$ are drawn. The computed results for Example 4.2 are listed in Tables 4.2–4.4 and Figures 4.4–4.5.

**Analysis steps:**

1. The GSVD induces simultaneous linear transformation of the two expression data sets $\{E_1, F_1\}$ and $\{E_2, F_2\}$ from 10020-genes × $n$-arrays spaces to the reduced $n$-genelets $n$-arraylets.
spaces. By transforming into the GSVD, the data sets \( \{E_1, F_1\} \) and \( \{E_2, F_2\} \) are represented by the diagonal non-negative matrices \( \{\Sigma_{E_1}, \Sigma_{F_1}\} \) and \( \{\Sigma_{E_2}, \Sigma_{F_2}\} \). We denote

\[
\langle k | \Sigma_{E_1} | l \rangle = \delta_{kl}, \quad \langle k | \Sigma_{F_1} | l \rangle = \delta_{kl},
\]

\[
\langle k | \Sigma_{E_2} | l \rangle = \delta_{kl}, \quad \langle k | \Sigma_{F_2} | l \rangle = \delta_{kl},
\]

for all \( 1 \leq k, l \leq n \), respectively and the antisymmetric angular distance between the data sets

\[
\theta_{lE_1,F_1} = \arctan(\alpha_l/\beta_l) - \pi/4, \quad \theta_{lE_2,F_2} = \arctan(\tilde{\alpha}_l/\tilde{\beta}_l) - \pi/4.
\]

We denote the GSVs of \( \{E_1, F_1\} \), \( \{E_2, F_2\} \) by

\[
\sigma \{E_1, F_1\} = \{\alpha_i, \beta_i\}, \quad \sigma \{E_2, F_2\} = \{\tilde{\alpha}_i, \tilde{\beta}_i\}, i = 1, \ldots, n.
\]

We first compute the values of \( \alpha_1 \) and \( \beta_n \).

1. If \( \alpha_1 \leq 1, \beta_n \leq 1 \), we compute \( \alpha_t \) for \( t \geq 2 \) and \( \beta_t \) for \( t \geq 2 \) till to \( \alpha_0 < 1, \beta_0 < 1 \). Then compute \( \alpha_j \) for \( t_0 \leq j \leq [n/2] \) till to \( \alpha_j = \sqrt{2/j} \).

2. If \( \alpha_j = \sqrt{2/j} \) and \( \alpha_{j+1} \leq \sqrt{2/j} \), then stop computing \( \alpha_j \) for \( j = j_0 + 2, \ldots, [n/2] \). If \( \alpha_j > \sqrt{2/j}, j = 1, 2, \ldots, [n/2] \), then compute \( \beta_k \) for \( k = n, n-1, \ldots, [n/2] + 1 \) till to \( \beta_k = \sqrt{2/k} \).

On the other hand, if \( \beta_k = \sqrt{2/k} \) and \( \beta_{k-1} \leq \sqrt{2/k} \), then stop computing \( \beta_k \) for \( k = n-2, \ldots, [n/2] - 1 \). If \( \beta_k > \sqrt{2/k} \) for \( k = n, n-1, \ldots, [n/2] - 1 \), then compute \( \alpha_j \) for \( j = 1, 2, \ldots, [n/2] \) till to \( \alpha_j = \sqrt{2/j} \).

4. We need find \( j \) for \( \alpha_j = 1, \sqrt{2/j} \) and \( k \) for \( \beta_k = 1, \sqrt{2/k} \).

5. Meanwhile, by Theorem 2.3 Lemma 2.3 and Lemma 2.4 we know that computing \( \alpha_j \) and \( \beta_k \) needs \( Q_j \) and \( G_j \) for \( j = 1, 2, \ldots, [n/2] \) and \( P_k \) and \( H_k \) for \( k = n, n-1, \ldots, [n/2] + 1 \). It refers to calculating only partial values of \( \alpha_j \) and \( \beta_k \), which costs much less.

By the above analysis steps we compute the following \( (\alpha_i, \beta_i) \) of \( \{E_1, F_1\} \) and \( (\tilde{\alpha}_i, \tilde{\beta}_i) \) of \( \{E_2, F_2\} \) for some \( i \) (see Tables 4.2-4.3). We give computation steps for \( \{E_1, F_1\} \) and \( \{E_2, F_2\} \) datasets for Example 4.1 as follows.

**Computation steps for \( \{E_1, F_1\} \) dataset:**

1. Firstly by using Algorithm 3.1 and svds we compute \( \alpha_1 = 0.81013202056465 \) and \( \beta_n = 0.718697912381140 \).

2. Then compute \( \beta_i, i = 127, \ldots, 121 \) and \( \beta_{121} = 0.70595228735626 \approx \sqrt{2} \) and continue to compute \( \beta_{120} = 0.705954228735626 < \sqrt{2} \).

3. Since \( \alpha_i \geq \alpha_{i+1} \) and \( \beta_i \leq \beta_{i+1} \) for \( i = 1, \ldots, 127 \), we compute \( \alpha_1 = 0.81013202056465 \) and \( \beta_1 = 0.586247481662121 \) and

\[
0.718697912381140 = \beta_{128} > \cdots > \beta_{122} = 0.711863678067525 > \beta_{121} = \sqrt{2} > \beta_{120} = 0.705954228735626 \geq \cdots \geq \beta_1 = 0.586247481662121.
\]

4. We need find \( j \) for \( \alpha_j = 1, \sqrt{2/j} \) and \( k \) for \( \beta_k = 1, \sqrt{2/k} \). From the above analysis we know that \( \beta_{121} = \sqrt{2} \) and for any \( j, \alpha_j \neq 1 \) and \( \beta_j \neq 1 \).
5. By $\alpha_i^2 + \beta_i^2 = 1$ for $i = 1, 120, \ldots, 128$ we can derive $\beta_1$ and $\alpha_i$ for $i = 1, 120, \ldots, 128$, and then compute $\theta_{lE_1,F_1} = \arctan(\alpha_l/\beta_l) - \pi/4 = 0$ for $l = 121$ (see Table 4.4 and Figure 4.4).

6. Therefore, we need compute $\alpha_1$ and $\beta_i$ for $i = 120, \ldots, 128$, which needs $Q_1, G_1, P_k, H_k$ for $k = 120 = n - 8, \ldots, 128 = n$ and the cost is much less by the proposed algorithms.

We then give computation steps for $\{E_2, F_2\}$ dataset for Example 4.2 as follows.

**Computation steps for $\{E_2, F_2\}$ dataset:**

1. Firstly, by using Algorithm 3.1 and svds we compute $\alpha_1 = 0.796382849891640$ and $\beta_n = \beta_{128} = 0.725245192340354$.

2. Then compute $\beta_i$ for $i = 127, \ldots, 118$ and $\beta_{118} = 0.707160154056073$ and continue to compute $\beta_{117} = 0.705954228735626 < \sqrt{2}/2$.

3. Since $\alpha_i \geq \alpha_{i+1}$ and $\beta_i \leq \beta_{i+1}$ for $i = 1, \ldots, 127$, we compute $\alpha_1 = 0.796382849891640$ and $\beta_1 = 0.604792821054011$ and

\[ 0.725245187463159 = \beta_{128} > \cdots > \beta_{119} = 0.707773720014017 > \beta_{118} \approx \frac{\sqrt{2}}{2} \]

\[ > \beta_{117} = 0.705546152277959 \geq \cdots \geq \beta_1 = 0.604792821054011. \]

4. We need find $j$ for $\alpha_j = 1, \sqrt{2}/2$ and $k$ for $\beta_k = 1, \sqrt{2}/2$. From the above analysis we know that $\beta_{118} = \sqrt{2}/2$ and for any $j, \alpha_j \neq 1$ and $\beta_j \neq 1$.

5. By $\alpha_i^2 + \beta_i^2 = 1$ for $i = 1, 117, \ldots, 128$ we derive that $\beta_1$ and $\alpha_i$ for $i = 1, 117, \ldots, 128$, and then compute $\theta_{lE_2,F_2} = \arctan(\tilde{\alpha}_l/\tilde{\beta}_l) - \pi/4$ for $l = 118$ (see Table 4.4 and Figure 4.5).

6. Therefore, we need compute $\alpha_1$ and $\beta_i$ for $i = 117, \ldots, 128$, which needs $Q_1, G_1, P_k, H_k$ for $k = 117 = n - 11, \ldots, 128 = n$ and also the cost is much less by the proposed algorithms.

We conclude the following results.

| Alg. | $i$   | $\alpha_i$       | $\beta_i$       |
|------|-------|------------------|------------------|
| Alg. 3.1 | i=1  | 0.810132020256465 | /                |
|       | i=128| /                | 0.718697912381140|
|       | i=127| /                | 0.717627620092847|
|       | i=126| /                | 0.717063249543065|
|       | i=125| /                | 0.714460025893188|
|       | i=124| /                | 0.713951890380495|
|       | i=123| /                | 0.712117595723366|
|       | i=122| /                | 0.711863780675258|
|       | i=121| /                | 0.707160154056073|
|       | i=120| /                | 0.705954228735626|
| svds | i=1  | 0.810111613718651 | /                |
|       | i=128| /                | 0.718688457942421|
|       | i=127| /                | 0.717629631885725|
|       | i=126| /                | 0.717060896558196|
|       | i=125| /                | 0.7144499213982706|
|       | i=124| /                | 0.713940747495409|
|       | i=123| /                | 0.712104173213587|
|       | i=122| /                | 0.711876696581871|
|       | i=121| /                | 0.707156111014042|
|       | i=120| /                | 0.705959880773849|

Table 4.2: Computed $(\alpha_i, \beta_i)$ for $\{E_1, F_1\}$ dataset for Example 4.2.
Table 4.3: Computed $(\tilde{\alpha}_i, \tilde{\beta}_i)$ for $\{E_2, F_2\}$ dataset for Example 4.2

| Alg.  | $i$   | $\tilde{\alpha}_i$ | $\tilde{\beta}_i$ |
|-------|-------|---------------------|-------------------|
| Alg. 3.1 | $i=1$ | 0.796382849891640   | /                 |
|       | $i=128$ | 0.725245192340354   | /                 |
|       | $i=127$ | 0.714661354527788   | /                 |
|       | $i=126$ | 0.71589075680548    | /                 |
|       | $i=125$ | 0.715628751189350   | /                 |
|       | $i=124$ | 0.713702581283016   | /                 |
|       | $i=123$ | 0.712809936437327   | /                 |
|       | $i=122$ | 0.710632060519230   | /                 |
|       | $i=121$ | 0.709521730665737   | /                 |
|       | $i=120$ | 0.707850060237633   | /                 |
|       | $i=119$ | 0.70773578361755    | /                 |
|       | $i=118$ | 0.70706687837273    | /                 |
|       | $i=117$ | 0.705546192429588   | /                 |
| svds  | $i=1$  | 0.796382814200499   | /                 |
|       | $i=128$ | 0.725245187463159   | /                 |
|       | $i=127$ | 0.71466140240871    | /                 |
|       | $i=126$ | 0.71589079555181    | /                 |
|       | $i=125$ | 0.71562873273613    | /                 |
|       | $i=124$ | 0.713702591000907   | /                 |
|       | $i=123$ | 0.71280935841902    | /                 |
|       | $i=122$ | 0.710632056740559   | /                 |
|       | $i=121$ | 0.70952177784081    | /                 |
|       | $i=120$ | 0.707850078689362   | /                 |
|       | $i=119$ | 0.7077320014017     | /                 |
|       | $i=118$ | 0.707066854930138   | /                 |
|       | $i=117$ | 0.705546152277959   | /                 |

- The computed values by Algorithm 3.1 and svds are almost the same as Algorithm 3.1.

- For $l = 121$, $\theta_{E_1,F_1} = 0$ and for $l = 118$, $\theta_{E_2,F_2} = 0$. This shows $E_1$ and $F_1$ from original and PA stimulation data sets have genelets of equal significance in $l = 121$-th genelet. $E_2$ and $F_2$ after knocking out SHP2 gene of mice without and with PA stimulation data sets have genelets of equal significance in $l = 118$-th genelet. Hence we can conclude that $E_1$ and $F_1$ from $E_1$-Sn and $F_1$-Sn data sets have genelets of equal significance in 121-th genelet and $E_2$ and $F_2$ from $E_2$-Sn data sets have genelets of equal significance in 118-th genelet. This is helpful and valuable for further experimental research when analyzing genelets of equal significance and no significance.

5 Concluding remarks

In this paper, from the aspect of small cost, high precision and practical applications, we give new formulations for computing arbitrary generalized singular value of Grassman matrix pair. Then by using these new formula models, we design the corresponding algorithms involving Newton’s method on Grassmann manifold and the MATLAB routine solver svds, which only result in smaller cost and higher precision. We apply the proposed algorithms in some numerical tests. Numerical
Table 4.4: $\theta_l\{E_1, F_1\}$ and $\theta_l\{E_2, F_2\}$ for Example 4.2

| $l$  | $\theta_l\{E_1, F_1\}$ | $l$  | $\theta_l\{E_2, F_2\}$ |
|------|------------------------|------|------------------------|
| i=128 | -0.016529700047278     | i=128 | -0.025992294349443     |
| i=127 | -0.014991647262518     | i=127 | -0.014759799050394     |
| i=126 | -0.01418160521803      | i=126 | -0.012500876385385     |
| i=125 | -0.010462462349320     | i=125 | -0.012125698075447     |
| i=124 | -0.009727915483202     | i=124 | -0.009371923296157     |
| i=123 | -0.007193397068787     | i=123 | -0.006097504074166     |
| i=122 | -0.006750101190428     | i=122 | -0.00499008671197      |
| i=121 | 0                      | i=121 | -0.003421112938104     |
| i=120 | 0.001628629810176      | i=120 | -0.001051708553944     |
|      |                        |      |                        |
|      |                        |      |                        |
|      |                        |      |                        |
|      |                        |      |                        |
|      |                        |      |                        |
| i=1  | 0.158979112164203      | i=1  | 0.135892471613447      |

Figure 4.4: $\theta_l\{E_1, F_1\}$ for Example 4.2

Tests are accessed to illustrate the efficiency of the proposed methods for computing arbitrary generalized singular value of Grassman matrix pair. The advantage of the new formula models and the corresponding algorithms is to compute several generalized singular values, which cost much less than making the whole generalized singular value decomposition.

References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre, Optimization Algorithms on Matrix Manifolds, Princeton University Press, Princeton, 2008.

[2] O. Alter, P. O. Brown, D. Botstein, Singular value decomposition for genome-wide expression data processing and modelling, Proc. Natl. Acad. Sci. USA, 97 (2000), pp. 10101–10106.
New model formula and methods for computing generalized singular values of GMP

Figure 4.5: $\theta \{E_2, F_2\}$ for Example 4.2.

[3] O. Alter, P. O. Brown, D. Botstein, Generalized singular decomposition for comparative analysis of genome-scale expression data sets of two different organisms, Proc. Natl. Acad. Sci. USA, 100 (2003), pp. 3351–3356.

[4] Z. Bai and J. W. Demmel, Computing the generalized singular value decomposition, SIAM J. Sci. Comput., 14 (1993), pp. 1464–1486.

[5] J. L. Barlow, Error analysis and implementation aspects of deferred correction for equality constrained least squares problems, SIAM J. Numer. Anal., 25 (1988), pp. 1340–1358.

[6] C. Davis, W. M. Kahan, and H. F. Weinberger, Norm-preserving dilations and their applications to optimal error bounds, SIAM J. Numer. Anal., 19 (1982), pp. 445–469.

[7] Z. Drmač, A tangent algorithm for computing the generalized singular value decomposition, SIAM J. Numer. Anal., 35 (1998), pp. 1804–1832.

[8] L. M. Ewerbring, F. T. Luk, Canonical correlations and generalized SVD: applications and new algorithms, J. Comput. Appl. Math., 27 (1989), pp. 37–52.

[9] G. H. Golub and W. Kahan, Calculating the singular values and pseudo-inverse of a matrix, J. SIAM Ser. B Numer. Anal., 2 (1965), pp. 205–224.

[10] G. H. Golub, C. F. Van Loan, Matrix Computations, 4th ed., Johns Hopkins University Press, Baltimore, 2013.

[11] R. A. Horn, C. R. Johnson, Matrix Analysis, 2nd ed., Cambridge University Press, Cambridge, 2013.

[12] L. K. Hua, Harmonic Analysis of Functions of Several Complex Variables in the Classical Domains, American Mathematical Society, Providence, 1963.

[13] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Res. Natl. Bur. Stand., 45 (1950), pp. 255–282.
[14] R. C. Li, Bounds on perturbations of generalized singular values and of associated subspaces, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 195–234.

[15] Q. K. Lu, The elliptic geometry of extended spaces, Acta Math. Sinica, 13 (1963), pp. 49–62.

[16] J. H. Manton, Optimization algorithms exploiting unitary constraints, IEEE Trans. Signal Process., 50 (2002), pp. 635–650.

[17] Y. Matsushima, Differentiable Manifolds, Marcel Dekker, New York, 1972.

[18] C. C. Paige, Computing the generalized singular value decomposition, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 1126–1146.

[19] C. C. Paige, M. A. Saunders, Towards a generalized singular value decomposition, SIAM J. Numer. Anal., 18 (1981), pp. 398–405.

[20] G. W. Stewart, Computing the CS-decomposition of a partitioned orthonormal matrix, Numer. Math., 40 (1982), pp. 297–306.

[21] J.-G. Sun, Perturbation analysis for the generalized singular value problem, SIAM J. Numer. Anal., 20 (1983), pp. 611–625.

[22] C. F. Van Loan, Generalizing the singular value decomposition, SIAM J. Numer. Anal., 13 (1976), pp. 76–83.

[23] C. F. Van Loan, Computing the CS and the generalized singular value decompositions, Numer. Math., 46 (1985), pp. 479–491.

[24] W.-W. Xu, H. K. Pang, W. Li, X. P. Huang, W. J. Sun, On the explicit expression of chordal metric between generalized singular values of Grassmann matrix pairs with applications, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 1547–1563.

[25] W.-W. Xu, W. Li, L. Zhu, X. P. Huang, The analytic solutions of a class of constrained matrix minimization and maximization problems with applications, SIAM J. Optim., 29 (2019), pp. 1657–1686.

[26] H. Zha, A numerical algorithm for computing restricted singular value decomposition of matrix triplets, Linear Algebra Appl., 168 (1992), pp. 1–25.

[27] Z. Zhao, Z. J. Bai, and X. Q. Jin, A Riemannian Newton algorithm for nonlinear eigenmannian Newton problems, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 752–774.

6 Appendix

Because the full data sets in Example 4.2 are big, we list a few lines from $E_1, F_1, E_2, F_2$ data sets in Tables 6.5–6.8. Full data sets will be attached as supplementary materials if needed.
New model formula and methods for computing generalized singular values of GMP

Table 6.5: $E_1$ dataset

| Gene-Symbol | $E_1 - S_1$ | $E_1 - S_2$ | $E_1 - S_3$ | \( \cdots \) |
|-------------|-------------|-------------|-------------|-------------|
| Dpm3        | 81.4726374728877 | 81.4726018003362 | 81.4727154775931 | \( \cdots \) |
| Gm15417     | 90.5801106506692 | 90.5795202525307 | 90.5787761181889 | \( \cdots \) |
| Pygo2       | 12.6975522059201 | 12.6988796188933 | 12.6993125974590 | \( \cdots \) |
| Shc1        | 91.3380167005621 | 91.3378745358918 | 91.3376629236392 | \( \cdots \) |
| Pbxip1      | 63.2360840051609 | 63.2367024531678 | 63.2364758975677 | \( \cdots \) |
| Pmvk        | 9.7538665579280  | 9.7552609353508  | 9.7550875896127  | \( \cdots \) |
| Kcm3        | 27.849605900937  | 27.850584828550  | 27.8511354102449 | \( \cdots \) |
| Adar        | 54.688323237317  | 54.6891680197768 | 54.6886325094756 | \( \cdots \) |
| Ube2q1      | 95.752427418996  | 95.7509942971894 | 95.7518196627301 | \( \cdots \) |
| She         | 96.490238284426  | 96.4892413334247 | 96.4893910841147 | \( \cdots \) |
| Gm19710     | 15.7606332242848 | 15.7631050234608 | 15.7608553343750 | \( \cdots \) |
| 1700094D03Rik | 97.0607956377947 | 97.0599299497393 | 97.0594282912228 | \( \cdots \) |
| 4933434E20Rik | 95.7170575264070 | 95.7173209697467 | 95.717053573553 | \( \cdots \) |

\( \cdots \) \( \cdots \) \( \cdots \) \( \cdots \)

Table 6.6: $F_1$ dataset

| Gene-Symbol | $F_1 - S_1$ | $F_1 - S_2$ | $F_1 - S_3$ | \( \cdots \) |
|-------------|-------------|-------------|-------------|-------------|
| Dpm3        | 81.4730879236440 | 81.4730419716306 | 81.4733975198905 | \( \cdots \) |
| Gm15417     | 90.579326384437 | 90.5790411296398 | 90.5798411038575 | \( \cdots \) |
| Pygo2       | 12.6994178526520 | 12.6991540372817 | 12.69910841147  | \( \cdots \) |
| Shc1        | 91.3383168331391 | 91.3378505277527 | 91.3376629236392 | \( \cdots \) |
| Pbxip1      | 63.2368278125429 | 63.2360364918807 | 63.2361050234608 | \( \cdots \) |
| Pmvk        | 9.75435665159052  | 9.75486146024052  | 9.7543834256857 | \( \cdots \) |
| Kcm3        | 27.849605900937  | 27.849228038682  | 27.8492926144878 | \( \cdots \) |
| Adar        | 54.6891748325179 | 54.6887342935505 | 54.6887102842456 | \( \cdots \) |
| Ube2q1      | 95.7509942971894 | 95.7518196627301 | 95.7519690164486 | \( \cdots \) |
| She         | 96.490238284426  | 96.4892413334247 | 96.4893910841147 | \( \cdots \) |
| Gm19710     | 15.7606332242848 | 15.7631050234608 | 15.7608553343750 | \( \cdots \) |
| 1700094D03Rik | 97.0599299497393 | 97.0594282912228 | 97.0594282912228 | \( \cdots \) |
| 4933434E20Rik | 95.7170575264070 | 95.7173209697467 | 95.717053573553 | \( \cdots \) |

\( \cdots \) \( \cdots \) \( \cdots \) \( \cdots \)
### Table 6.7: $E_2$ dataset

| Gene-Symbol | $F_1 - S_1$ | $F_1 - S_2$ | $F_1 - S_3$ | ... |
|-------------|-------------|-------------|-------------|-----|
| Dpm3        | 81.4523098512597 | 81.4629821583036 | 81.4745727961033 | ... |
| Gm15417     | 90.5779819953193 | 90.5848577932900 | 90.5906502919661 | ... |
| Pygo2       | 12.6974035147009 | 12.6873751483440 | 12.7044923008073 | ... |
| Shc1        | 91.3286343984840 | 91.3251461714906 | 91.3490834650614 | ... |
| Pbxip1      | 63.2309324317487 | 63.2334546316634 | 63.2183204836416 | ... |
| Pmvk        | 9.7503406488495  | 9.7469591076062 | 9.7622936309787 | ... |
| Kcnm3       | 27.8621314570003 | 27.8559191575373 | 27.8528545813737 | ... |
| Adar        | 54.7078492202499 | 54.6885058093600 | 54.6830922478760 | ... |
| Ube2q1      | 95.7501045170701 | 95.7625720815543 | 95.7713725989043 | ... |
| She         | 96.4900197059825 | 96.4971930401927 | 96.4960354367868 | ... |
| Gm19710     | 15.7624053853472 | 15.7593724639648 | 15.7652870075486 | ... |
| 170094D03Rik| 97.0562110151379 | 97.0573160527630 | 97.051754414638 | ... |
| 4933434E20Rik| 95.7235525356460 | 95.719450298556 | 95.7137803601465 | ... |

### Table 6.8: $F_2$ dataset

| Gene-Symbol | $F_2$-S1 | $F_2$-S2 | $F_2$-S3 | ... |
|-------------|----------|----------|----------|-----|
| Dpm3        | 81.4817402134289 | 81.4749725312774 | 81.481241209804 | ... |
| Gm15417     | 90.5840967987585 | 90.5899209907963 | 90.5746909609589 | ... |
| Pygo2       | 12.6928679920493 | 12.7027778119699 | 12.7076685037737 | ... |
| Shc1        | 91.3441570447496 | 91.3397332006277 | 91.3264759805061 | ... |
| Pbxip1      | 63.2375890545070 | 63.2321703530508 | 63.2410869112732 | ... |
| Pmvk        | 9.7527488967974  | 9.7541205844562 | 9.7456474253063 | ... |
| Kcnm3       | 27.8676517195286 | 27.8522488517431 | 27.8484309535345 | ... |
| Adar        | 54.6778922856374 | 54.6881209028444 | 54.6752501090303 | ... |
| Ube2q1      | 95.7382577903967 | 95.7358047141356 | 95.748830100979 | ... |
| She         | 96.4874304231390 | 96.5078630839640 | 96.4794574795017 | ... |
| Gm19710     | 15.7754836985419 | 15.7497320889512 | 15.7634036450606 | ... |
| 170094D03Rik| 97.0546341778293 | 97.0583580985808 | 97.045437736231 | ... |
| 4933434E20Rik| 95.7202553317885 | 95.7103795510779 | 95.708058343294 | ... |

...